

```

MMM          MMM          AAAAAAAAAA          IIIIIIIIIII          LLL
MMM          MMM          AAAAAAAAAA          IIIIIIIIIII          LLL
MMM          MMM          AAAAAAAAAA          IIIIIIIIIII          LLL
MMMMMMMM    MMMMMMM    AAA          AAA          III          LLL
MMMMMMMM    MMMMMMM    AAA          AAA          III          LLL
MMMMMMMM    MMMMMMM    AAA          AAA          III          LLL
MMM          MMM          AAA          AAA          III          LLL
MMM          MMM          AAA          AAA          III          LLL
MMM          MMM          AAA          AAA          III          LLL
MMM          MMM          AAA          AAA          III          LLL
MMM          MMM          AAA          AAA          III          LLL
MMM          MMM          AAA          AAA          III          LLL
MMM          MMM          AAAAAAAAAAAAAAAAAA          III          LLL
MMM          MMM          AAAAAAAAAAAAAAAAAA          III          LLL
MMM          MMM          AAAAAAAAAAAAAAAAAA          III          LLL
MMM          MMM          AAA          AAA          III          LLL
MMM          MMM          AAA          AAA          III          LLL
MMM          MMM          AAA          AAA          III          LLL
MMM          MMM          AAA          AAA          III          LLL
MMM          MMM          AAA          AAA          IIIIIIIIIII          LLLLLLLLLLLLLLLLLL
MMM          MMM          AAA          AAA          IIIIIIIIIII          LLLLLLLLLLLLLLLLLL
MMM          MMM          AAA          AAA          IIIIIIIIIII          LLLLLLLLLLLLLLLLLL

```

MA
VO

.....

:

```

CCCCCCCCC MM MM DDDDDDDDD SSSSSSSSS
CCCCCCCCC MM MM DDDDDDDDD SSSSSSSSS
CC M M M M DD DD SS
CC M M M M DD DD SS
CC MM MM MM DD DD SS
CC MM MM MM DD DD SS
CC MM MM MM DD DD SSSSSSS
CC MM MM MM DD DD SSSSSSS
CC MM MM MM DD DD SS
CC MM MM MM DD DD SS
CC MM MM MM DD DD SS
CC MM MM MM DD DD SS
CCCCCCCCC MM MM DDDDDDDDD SSSSSSSSS
CCCCCCCCC MM MM DDDDDDDDD SSSSSSSSS

```

```

LL          IIIIII          SSSSSSSS
LL          IIIIII          SSSSSSSS
LL          II             SS
LL          II             SS
LL          II             SS
LL          II             SS
LL          II             SSSSSS
LL          II             SSSSSS
LL          II             SS
LL          II             SS
LL          II             SS
LL          II             SS
LLLLLLLLLLLL IIIIII          SSSSSSSS
LLLLLLLLLLLL IIIIII          SSSSSSSS

```



```
1 0001 0 MODULE MAIL$CMDS( %TITLE 'VAX/VMS PERSONAL MAIL UTILITY COMMANDS'
2 0002 0 IDENT = 'V04-000' ) =
3 0003 1 BEGIN
4 0004 1
5 0005 1
6 0006 1 *****
7 0007 1 *
8 0008 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
9 0009 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
10 0010 1 * ALL RIGHTS RESERVED.
11 0011 1 *
12 0012 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
13 0013 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
14 0014 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
15 0015 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
16 0016 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
17 0017 1 * TRANSFERRED.
18 0018 1 *
19 0019 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
20 0020 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
21 0021 1 * CORPORATION.
22 0022 1 *
23 0023 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
24 0024 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
25 0025 1 *
26 0026 1 *
27 0027 1 *****
28 0028 1
29 0029 1
30 0030 1 ++
31 0031 1 FACILITY: VAX/VMS MAIL UTILITY
32 0032 1
33 0033 1 ABSTRACT: Interactive command processing.
34 0034 1
35 0035 1 ENVIRONMENT: NATIVE/USER MODE
36 0036 1
37 0037 1 AUTHOR: Benn Schreiber, CREATION DATE: 22-Dec-1982
38 0038 1
39 0039 1 MODIFIED BY:
40 0040 1
41 0041 1
42 0042 1 V03-015 ROP0028 Robert Posniak 23-JUL-1984
43 0043 1 Make sure do not point past the end of the
44 0044 1 current folder with the DELETE command. Check
45 0045 1 and report error status after deleting message.
46 0046 1
47 0047 1 V03-014 BLS0272 Benn Schreiber 18-FEB-1984
48 0048 1 Disable ctrl/c handler around help. Use LIB$FIND_IMAGE_SYMBOL
49 0049 1 rather than UTIL$ version.
50 0050 1
51 0051 1 V03-013 BLS0263 Benn Schreiber 22-Jan-1984
52 0052 1 Don't allow SPAWN if captive user. Implement COMPRESS
53 0053 1 command to convert mail file.
54 0054 1
55 0055 1 V03-012 BLS0255 Benn Schreiber 28-Dec-1983
56 0056 1 Convert byte flags to a flags longword. /NEW implies
57 0057 1 the default mail file. Make /EDIT work on sequential
```

```

: 58      0058 1 1
: 59      0059 1
: 60      0060 1
: 61      0061 1
: 62      0062 1
: 63      0063 1
: 64      0064 1
: 65      0065 1
: 66      0066 1
: 67      0067 1
: 68      0068 1
: 69      0069 1
: 70      0070 1
: 71      0071 1
: 72      0072 1
: 73      0073 1
: 74      0074 1
: 75      0075 1
: 76      0076 1
: 77      0077 1
: 78      0078 1
: 79      0079 1
: 80      0080 1
: 81      0081 1
: 82      0082 1
: 83      0083 1
: 84      0084 1
: 85      0085 1
: 86      0086 1
: 87      0087 1
: 88      0088 1
: 89      0089 1
: 90      0090 1
: 91      0091 1
: 92      0092 1
: 93      0093 1
: 94      0094 1
: 95      0095 1
: 96      0096 1
: 97      0097 1 1--

files. Make first, last, search work on sequential
files. Spawn and attach do a purgws.

V03-011 BLS0246      Benn Schreiber      28-Nov-1983
Add defaults of 'TODAY' to /BEFORE and /SINCE. Add
/EDIT qualifier to BACK, CURRENT, FIRST, NEXT, LAST,
and READ commands. Add /CONFIRM to MOVE, FILE, COPY.

V03-010 BLS0235      Benn Schreiber      23-Aug-1983
Fix defaulting of folder selection if file already open.
Clear garbage bits in return status passed as bit flag.

V03-009 BLS0227      Benn Schreiber      29-Jun-1983
Add routine to clear dir_flag and dir_mode. Correct
method of deleting messages with MOVE, FILE commands.

V03-008 BLS0226      Benn Schreiber      27-Jun-1983
Make DIR/START=999 give a full screen listing

V03-007 BLS0225      Benn Schreiber      17-Jun-1983
Make SEARCH command open file and select if not already done.

V03-006 BLS0222      Benn Schreiber      15-May-1983
Disable CTRL/C ast around spawning/attaching to subprocess

V03-005 BLS0221      Benn Schreiber      5-May-1983
Fix SPAWN/PROCESS=

V03-004 BLS0220      Benn Schreiber      30-Apr-1983
Interlock with PURGE, PURGE/RECLAIM to prevent extaneous
messages.

V03-003 BLS0217      Benn Schreiber      14-Apr-1983
Add SPAWN, ATTACH

V03-002 BLS0214      Benn Schreiber      27-Mar-1983
Add DELETE/ALL command. Cause FIRST and LAST commands to
open mail file and read default messages if not done yet.
```


99	0098	1	TABLE OF CONTENTS:	
100	0099	1		
101	0100	1		
102	0101	1		
103	0102	1		
104	0103	1	INCLUDE FILES:	
105	0104	1		
106	0105	1	LIBRARY 'SYSS\$LIBRARY:STARLET';	
107	0106	1	REQUIRE 'SRC\$:MAILREQ';	
108	0252	1	LIBRARY 'LIB\$:MAILDEF';	
109	0253	1		
110	0254	1	EXTERNAL ROUTINE	
111	0255	1	CLIS\$GET_VALUE: ADDRESSING_MODE(GENERAL),	Get value for qualifier
112	0256	1	CLIS\$PRESENT: ADDRESSING_MODE(GENERAL),	Test if qualifier present
113	0257	1	LIB\$ATTACH,	Attach to a subprocess
114	0258	1	LIB\$CVT_DTB: ADDRESSING_MODE(GENERAL),	Convert decimal to binary
115	0259	1	LIB\$CVT_TIME: ADDRESSING_MODE(GENERAL),	Convert date/time
116	0260	1	LIB\$SPAWN,	Spawn a subprocess
117	0261	1	MAIL\$CLOSE_FILE,	Close open msg file
118	0262	1	MAIL\$CMD_ERASE,	Erase the screen
119	0263	1	MAIL\$CMD_SHOW_NEW,	Show new message count
120	0264	1	MAIL\$COMPRESS,	Compress a mail file
121	0265	1	MAIL\$COPY_MSGS,	Copy messages to a file
122	0266	1	MAIL\$DELETE_I_MSG,	Delete current ISAM message
123	0267	1	MAIL\$DELETE_MSG,	Delete current message
124	0268	1	MAIL\$DIRECTORY,	Display summary of messages
125	0269	1	MAIL\$DISABLE_CTRL_C,	Disable CTRL/C trap
126	0270	1	MAIL\$ENABLE_CTRL_C,	Enable CTRL/C trap
127	0271	1	MAIL\$EDIT,	Edit a message
128	0272	1	MAIL\$EDIT_READ_MSG,	Edit/read a message
129	0273	1	MAIL\$DIR_FOLDER,	Display listing of folders
130	0274	1	MAIL\$EMPTY_WASTEBASKET,	Delete msgs from wastebasket
131	0275	1	MAIL\$FILE_I_MSG,	File current message
132	0276	1	MAIL\$FIND_I_MSG,	Find ISAM message
133	0277	1	MAIL\$FIND_MSG,	Find a message
134	0278	1	MAIL\$GET_INPUT,	Get from sys\$input
135	0279	1	MAIL\$GET_MSG,	Get record of current message
136	0280	1	MAIL\$GET_I_MSG,	Get record from ISAM message
137	0281	1	MAIL\$INFO_I_MSG,	Get ISAM message info
138	0282	1	MAIL\$INFO_MSG,	Get message information
139	0283	1	MAIL\$INFO_FOLDER,	Get folder info
140	0284	1	MAIL\$OPEN_FILE,	Open a mail file
141	0285	1	MAIL\$PRINT,	Print a message
142	0286	1	MAIL\$PUT_OUTPUT,	Put to sys\$output
143	0287	1	MAIL\$READ,	Display a message
144	0288	1	MAIL\$RECLAIM,	Convert/reclaim isam file
145	0289	1	MAIL\$REMOVE_I_MSG,	Remove message from current
146	0290	1	MAIL\$REWIND_MSG,	Rewind current message
147	0291	1	MAIL\$REWIND_I_MSG,	Rewind ISAM current message
148	0292	1	MAIL\$SELECT_MSGS,	Select ISAM messages
149	0293	1	MAIL\$STRIP_QUOTES,	Strip quote marks from string
150	0294	1	MAIL\$TEST_SUBJ,	Test if SUBJ: line present
151	0295	1	MAIL\$UPCASE,	Convert string to upper case
152	0296	1	MAIL\$UPCASE_Q,	Ditto, but not inside quotes
153	0297	1	OTSS\$COPY_DX: ADDRESSING_MODE(GENERAL),	String copy desc to desc
154	0298	1	STR\$COPY_DX: ADDRESSING_MODE(GENERAL),	String copy
155	0299	1	STR\$FREE_T_DX: ADDRESSING_MODE(GENERAL),	Deallocate dynamic string

```
156 0300 1 LIB$FIND_IMAGE_SYMBOL; ! Load image, find symbol value
157 0301 1
158 0302 1 EXTERNAL
159 0303 1 MAIL$GL_FLAGS : $BBLOCK, ! Control flags
160 0304 1 MAIL$B_MSGTRAN : BYTE, ! ISAM getmsg transition
161 0305 1 MAIL$SL_OUTLINES, ! Number of lines on page
162 0306 1 MAIL$SL_SYMAIBUF : REF $BBLOCK, ! Pointer to VMSMAIL.DAT buffer
163 0307 1 MAIL$SL_MSG_NUM : LONG, ! Current message number
164 0308 1 MAIL$SL_MSG_RADR : LONG, ! Current message record addr
165 0309 1 MAIL$W_MSG_RSIZ : WORD, ! Current message record size
166 0310 1 MAIL$SL_MSGBUFSIZ, ! Size of seq. file msg buf
167 0311 1 MAIL$SL_MSG_STS : LONG, ! Current message status
168 0312 1 MAIL$Q_CURFOLDER : $BBLOCK, ! Current folder name
169 0313 1 MAIL$Q_WASTEDESC : $BBLOCK, ! Wastebasket folder name
170 0314 1 MAIL$Q_MSGNAME : $BBLOCK, ! message file name
171 0315 1 MAIL$Q_MSGTYPE : $BBLOCK, ! Default message file type
172 0316 1 MAIL$Q_RD_DESC : $BBLOCK, ! Descriptor of current file
173 0317 1 MAIL$SL_NMSG, ! Number of ISAM messages
174 0318 1 MAIL$SL_CURMSG : REF $BBLOCK, ! Pointer to current msg block
175 0319 1 MAIL$SL_MSGTBL : REF VECTOR[,LONG]; ! Pointer to message table
176 0320 1
177 0321 1 GLOBAL
178 0322 1 MAIL$A_OUTPUT_HELP; ! Address of LBR$OUTPUT_HELP
179 0323 1
180 0324 1 OWN
181 0325 1 SEARCH_STRING_BUFF : $BBLOCK[128], ! Current search string buffer
182 0326 1 DIR_FLAG : BYTE, ! DIRECTORY last command flag
183 0327 1 ! and there is more DIR to go
184 0328 1 DIRFULL_FLAG : BYTE, ! DIR/FULL last command
185 0329 1 DIR_MODE : BYTE, ! 1=DIRECTORY, 0=READ
186 0330 1 SELECT_DATA : VECTOR[5,LONG], ! Used to parse selection stuff
187 0331 1 SELECT_FROM : DYN_DESC_DECL, ! ditto
188 0332 1 SELECT_NEW : LONG, ! ditto
189 0333 1 SELECT_REPLIED : LONG, ! ditto
190 0334 1 DYN_DESC : DYN_DESC_DECL, ! Dynamic descriptors
191 0335 1 DYN_DESC_2 : DYN_DESC_DECL,
192 0336 1 DYN_DESC_3 : DYN_DESC_DECL,
193 0337 1 COMMAND_DESC : DYN_DESC_DECL,
194 0338 1 SEARCH_STRING : $BBLOCK[8] ! Current search string desc
195 0339 1 PRESET([DSC$A_POINTER] = SEARCH_STRING_BUFF);
196 0340 1
197 0341 1 BIND
198 0342 1 SELECT_FLAGS = SELECT_DATA[0] : $BBLOCK; ! Message selection flags
199 0343 1
200 0344 1 !
201 0345 1 ! Declare static strings for qualifier names
202 0346 1 !
203 P 0347 1 SD(
204 P 0348 1 'ALL',
205 P 0349 1 'BEFORE',
206 P 0350 1 'CONFIRM',
207 P 0351 1 'COPIES',
208 P 0352 1 'DELNUM',
209 P 0353 1 'HEADER',
210 P 0354 1 'INPUT',
211 P 0355 1 'FDL',
212 P 0356 1 'FILE',
```



```
: 213 P 0357 1 'FOLDER_NAME',
: 214 P 0358 1 'FOLDER',
: 215 P 0359 1 'FROM',
: 216 P 0360 1 'FULL',
: 217 P 0361 1 'LOG',
: 218 P 0362 1 'LOGICAL_NAMES',
: 219 P 0363 1 'NOTIFY',
: 220 P 0364 1 'OUTPUT',
: 221 P 0365 1 'P1',
: 222 P 0366 1 'PARENT',
: 223 P 0367 1 'PRINT',
: 224 P 0368 1 'PROCESS',
: 225 P 0369 1 'QUEUE',
: 226 P 0370 1 'RECLAIM',
: 227 P 0371 1 'STATISTICS',
: 228 P 0372 1 'SYMBOLS',
: 229 P 0373 1 'TODAY',
: 230 P 0374 1 'REPLF',
: 231 P 0375 1 'EDIT',
: 232 P 0376 1 'HELPKEYS',
: 233 P 0377 1 'NEW',
: 234 P 0378 1 'READ1',
: 235 P 0379 1 'READ2',
: 236 P 0380 1 'SEARCH1',
: 237 P 0381 1 'SEND1',
: 238 P 0382 1 'SINCE',
: 239 P 0383 1 'START',
: 240 P 0384 1 'SUBJECT',
: 241 P 0385 1 'NEXT',
: 242 P 0386 1 'SETF1',
: 243 P 0387 1 'SETN1',
: 244 P 0388 1 'WAIT',
: 245 0389 1 'TOLIST');
: 246 0390 1
: 247 0391 1 GLOBAL BIND
: 248 0392 1 MAIL$SD_SETF1 = SD_SETF1,
: 249 0393 1 MAIL$SD_SETN1 = SD_SETN1;
: 250 0394 1
: 251 0395 1 Define shared messages as local
: 252 0396 1
: 253 P 0397 1 $SHR_MSGDEF(MAIL,126,LOCAL,
: 254 P 0398 1 (BADVALUE,ERROR),
: 255 P 0399 1 (CREATED,SUCCESS),
: 256 0400 1 (INVQUAVAL,ERROR));
```

!For other modules

```
: 258      0401 1 GLOBAL ROUTINE MAIL$OPEN_IF =
: 259      0402 1 ++
: 260      0403 1 FUNCTIONAL DESCRIPTION:
: 261      0404 1
: 262      0405 1   Open the default mail file if no mail file is open
: 263      0406 1
: 264      0407 1   INPUTS:
: 265      0408 1
: 266      0409 1       None
: 267      0410 1
: 268      0411 1   IMPLICIT INPUTS:
: 269      0412 1
: 270      0413 1       MAIF_V_FILOPN = 0 if no file open, 1 if file open
: 271      0414 1
: 272      0415 1   --
: 273      0416 2 BEGIN
: 274      0417 2
: 275      0418 2 LOCAL
: 276      0419 2     DESC : VECTOR[2, LONG];
: 277      0420 2
: 278      0421 2 IF .MAIL$GL_FLAGS[MAIF_V_FILOPN]
: 279      0422 2     THEN RETURN TRUE;
: 280      0423 2
: 281      0424 2 DESC[0] = DESC[1] = 0;
: 282      0425 2 RETURN MAIL$OPEN_FILE(DESC)
: 283      0426 1 END;
```

.TITLE MAIL\$CMDS VAX/VMS PERSONAL MAIL UTILITY COMMAND

.IDENT \V04-000\

.PSECT \$CODE\$,NOWRT,2

			4C	4C	41	00000	P.AAB:	.ASCII	\ALL\	:
						00003		.BLKB	1	:
					00000003	00004	P.AAA:	.LONG	3	:
					00000000	00008		.ADDRESS	P.AAB	:
45	52	4F	46	45	42	0000C	P.AAD:	.ASCII	\BEFORE\	:
						00012		.BLKB	2	:
					00000006	00014	P.AAC:	.LONG	6	:
					00000000	00018		.ADDRESS	P.AAD	:
4D	52	49	46	4E	4F	43	0001C	P.AAF:	.ASCII	\CONFIRM\
						00023		.BLKB	1	:
					00000007	00024	P.AAE:	.LONG	7	:
					00000000	00028		.ADDRESS	P.AAF	:
53	45	49	50	4F	43	0002C	P.AAH:	.ASCII	\COPIES\	:
						00032		.BLKB	2	:
					00000006	00034	P.AAG:	.LONG	6	:
					00000000	00038		.ADDRESS	P.AAH	:
4D	55	4E	4C	45	44	0003C	P.AAJ:	.ASCII	\DELNUM\	:
						00042		.BLKB	2	:
					00000006	00044	P.AAI:	.LONG	6	:
					00000000	00048		.ADDRESS	P.AAJ	:
52	45	44	41	45	48	0004C	P.AAL:	.ASCII	\HEADER\	:
						00052		.BLKB	2	:
					00000006	00054	P.AAK:	.LONG	6	:

[illegible]

						00000000'	0013C		.ADDRESS P.ABP	:				
	45	55	45	55	51		00140	P.ABR:	.ASCII \QUEUE\	:				
							00145		.BLKB 3	:				
						00000005	00148	P.ABQ:	.LONG 5	:				
						00000000'	0014C		.ADDRESS P.ABR	:				
	4D	49	41	4C	43	45	52	00150	P.ABT:	.ASCII \RECLAIM\	:			
								00157	.BLKB 1	:				
						00000007	00158	P.ABS:	.LONG 7	:				
						00000000'	0015C		.ADDRESS P.ABT	:				
	53	43	49	54	53	49	54	41	54	53	00160	P.ABV:	.ASCII \STATISTICS\	:
											0016A	.BLKB 2	:	
						0000000A	0016C	P.ABU:	.LONG 10	:				
						00000000'	00170		.ADDRESS P.ABV	:				
	53	4C	4F	42	4D	59	53	00174	P.ABX:	.ASCII \SYMBOLS\	:			
								0017B	.BLKB 1	:				
						00000007	0017C	P.ABW:	.LONG 7	:				
						00000000'	00180		.ADDRESS P.ABX	:				
	59	41	44	4F	54			00184	P.ABZ:	.ASCII \TODAY\	:			
								00189	.BLKB 3	:				
						00000005	0018C	P.ABY:	.LONG 5	:				
						00000000'	00190		.ADDRESS P.ABZ	:				
	46	4C	50	45	52			00194	P.ACB:	.ASCII \REPLF\	:			
								00199	.BLKB 3	:				
						00000005	0019C	P.ACA:	.LONG 5	:				
						00000000'	001A0		.ADDRESS P.ACB	:				
		54	49	44	45			001A4	P.ACD:	.ASCII \EDIT\	:			
						00000004	001A8	P.ACC:	.LONG 4	:				
						00000000'	001AC		.ADDRESS P.ACD	:				
	53	59	45	4B	50	4C	45	48	001B0	P.ACF:	.ASCII \HELPKEYS\	:		
						00000008	001B8	P.ACE:	.LONG 8	:				
						00000000'	001BC		.ADDRESS P.ACF	:				
						57	45	4E	001C0	P.ACH:	.ASCII \NEW\	:		
									001C3	.BLKB 1	:			
						00000003	001C4	P.ACG:	.LONG 3	:				
						00000000'	001C8		.ADDRESS P.ACH	:				
	31	44	41	45	52			001CC	P.ACJ:	.ASCII \READ1\	:			
								001D1	.BLKB 3	:				
						00000005	001D4	P.ACI:	.LONG 5	:				
						00000000'	001D8		.ADDRESS P.ACJ	:				
	32	44	41	45	52			001DC	P.ACL:	.ASCII \READ2\	:			
								001E1	.BLKB 3	:				
						00000005	001E4	P.ACK:	.LONG 5	:				
						00000000'	001E8		.ADDRESS P.ACL	:				
	31	48	43	52	41	45	53	001EC	P.ACN:	.ASCII \SEARCH1\	:			
								001F3	.BLKB 1	:				
						00000007	001F4	P.ACM:	.LONG 7	:				
						00000000'	001F8		.ADDRESS P.ACN	:				
	31	44	4E	45	53			001FC	P.ACP:	.ASCII \SEND1\	:			
								00201	.BLKB 3	:				
						00000005	00204	P.ACO:	.LONG 5	:				
						00000000'	00208		.ADDRESS P.ACP	:				
	45	43	4E	49	53			0020C	P.ACR:	.ASCII \SINCE\	:			
								00211	.BLKB 3	:				
						00000005	00214	P.ACQ:	.LONG 5	:				
						00000000'	00218		.ADDRESS P.ACR	:				
	54	52	41	54	53			0021C	P.ACT:	.ASCII \START\	:			
								00221	.BLKB 3	:				


```
00000005 00224 P.ACS: .LONG 5
00000000' 00228 .ADDRESS P.ACT
54 43 45 4A 42 55 53 0022C P.ACV: .ASCII \SUBJECT\
00233 .BLKB 1
00000007 00234 P.ACU: .LONG 7
00000000' 00238 .ADDRESS P.ACV
31 54 58 45 4E 0023C P.ACX: .ASCII \NEXT\
00000004' 00240 P.ACW: .LONG 4
00000000' 00244 .ADDRESS P.ACX
31 46 54 45 53 00248 P.ACZ: .ASCII \SETF1\
0024D .BLKB 3
00000005 00250 P.ACY: .LONG 5
00000000' 00254 .ADDRESS P.ACZ
31 4E 54 45 53 00258 P.ADB: .ASCII \SETN1\
0025D .BLKB 3
00000005 00260 P.ADA: .LONG 5
00000000' 00264 .ADDRESS P.ADB
54 49 41 57 00268 P.ADD: .ASCII \WAIT\
00000004' 0026C P.ADC: .LONG 4
00000000' 00270 .ADDRESS P.ADD
54 53 49 4C 4F 54 00274 P.ADF: .ASCII \TOLIST\
0027A .BLKB 2
00000006 0027C P.ADE: .LONG 6
00000000' 00280 .ADDRESS P.ADF
```

.PSECT \$OWNS,NOEXE,2

```
00000 SEARCH_STRING_BUFF:
00080 DIR_FLAG: .BLKB 128
00081 DIRFULL_FLAG: .BLKB 1
00082 DIR_MODE: .BLKB 1
00083 .BLKB 1
00084 SELECT_DATA: .BLKB 20
00# 00098 SELECT_FROM: .BYTE 0[2]
02 0E 0009A .BYTE 14, 2
0009C .BLKB 4
000A0 SELECT_NEW: .BLKB 4
000A4 SELECT_REPLIED: .BLKB 4
00# 000AB DYN_DESC: .BYTE 0[2]
02 0E 000AA .BYTE 14, 2
000AC .BLKB 4
00# 000B0 DYN_DESC 2: .BYTE 0[2]
02 0E 000B2 .BYTE 14, 2
000B4 .BLKB 4
00# 000B8 DYN_DESC 3: .BYTE 0[2]
02 0E 000BA .BYTE 14, 2
```

```
00# 000BC .BLKB 4
00# 000C0 COMMAND_DESC:
02 0E 000C2 .BYTE 0[2]
000C4 .BYTE 14, 2
00# 000C8 SEARCH_STRING:
00000000' 000CC .BYTE 0[4]
               .ADDRESS SEARCH_STRING_BUFF
               .PSECT $GLOBAL$,NOEXE,2
00000 MAIL$A_OUTPUT_HELP::
               .BLKB 4
```

SELECT_FLAGS=	SELECT_DATA
SD_ALL=	P.AAA
SD_BEFORE=	P.AAC
SD_CONFIRM=	P.AAE
SD_COPIES=	P.AAG
SD_DELNUM=	P.AAI
SD_HEADER=	P.AAK
SD_INPUT=	P.AAM
SD_FDL=	P.AAO
SD_FILE=	P.AAQ
SD_FOLDER_NAME=	P.AAS
SD_FOLDER=	P.AAU
SD_FROM=	P.AAW
SD_FULL=	P.AAY
SD_LOG=	P.ABA
SD_LOGICAL_NAMES=	P.ABC
SD_NOTIFY=	P.ABE
SD_OUTPUT=	P.ABG
SD_P1=	P.ABI
SD_PARENT=	P.ABK
SD_PRINT=	P.ABM
SD_PROCESS=	P.ABO
SD_QUEUE=	P.ABQ
SD_RECLAIM=	P.ABS
SD_STATISTICS=	P.ABU
SD_SYMBOLS=	P.ABW
SD_TODAY=	P.ABY
SD_REPLF=	P.ACA
SD_EDIT=	P.ACC
SD_HELPKEYS=	P.ACE
SD_NEW=	P.ACG
SD_READ1=	P.ACI
SD_READ2=	P.ACK
SD_SEARCH1=	P.ACM
SD_SEND1=	P.ACO
SD_SINCE=	P.ACQ
SD_START=	P.ACS
SD_SUBJECT=	P.ACU
SD_NEXT=	P.ACW
SD_SETF1=	P.ACY
SD_SETN1=	P.ADA
SD_WAIT=	P.ADC
SD_TOLIST=	P.ADE


```
MAIL$SD_SETF1== P.ACY
MAIL$SD_SETN1== P.ADA
.EXTRN CLISGET VALUE, CLISPRESENT
.EXTRN LIB$ATTACH, LIB$CVT DTB
.EXTRN LIB$CVT TIME, LIB$SPAWN
.EXTRN MAIL$CLOSE FILE
.EXTRN MAIL$CMD ERASE, MAIL$CMD_SHOW_NEW
.EXTRN MAIL$COMPRESS, MAIL$COPY_MSGS
.EXTRN MAIL$DELETE_I MSG
.EXTRN MAIL$DELETE_MSG
.EXTRN MAIL$DIRECTORY, MAIL$DISABLE_CTRLC
.EXTRN MAIL$ENABLE_CTRLC
.EXTRN MAIL$EDIT, MAIL$EDIT_READ_MSG
.EXTRN MAIL$DIR FOLDER
.EXTRN MAIL$EMPTY WASTEBASKET
.EXTRN MAIL$FILE_I MSG
.EXTRN MAIL$FIND_I MSG
.EXTRN MAIL$FIND_MSG, MAIL$GET_INPUT
.EXTRN MAIL$GET_MSG, MAIL$GET_I_MSG
.EXTRN MAIL$INFO_I MSG
.EXTRN MAIL$INFO_MSG, MAIL$INFO FOLDER
.EXTRN MAIL$OPEN_FILE, MAIL$PRINT
.EXTRN MAIL$PUT OUTPUT
.EXTRN MAIL$READ, MAIL$RECLAIM
.EXTRN MAIL$REMOVE_I MSG
.EXTRN MAIL$REWIND_MSG
.EXTRN MAIL$REWIND_I MSG
.EXTRN MAIL$SELECT_MSGS
.EXTRN MAIL$STRIP_QUOTES
.EXTRN MAIL$TEST SUBJ, MAIL$UPCASE
.EXTRN MAIL$UPCASE_Q, OTS$SCOPY DXDX
.EXTRN STR$COPY_DX, STR$FREE1_DX
.EXTRN LIB$FIND_IMAGE_SYMBOL
.EXTRN MAIL$GL_FLAGS, MAIL$B_MSGTRAN
.EXTRN MAIL$L_OUTLINES
.EXTRN MAIL$L_SYMAIBUF
.EXTRN MAIL$L_MSG_NUM, MAIL$L_MSG_RADR
.EXTRN MAIL$W_MSG_RSIZ
.EXTRN MAIL$L_MSGBUFSIZ
.EXTRN MAIL$L_MSG_STS, MAIL$Q_CURFOLDER
.EXTRN MAIL$Q_WASTEDESC
.EXTRN MAIL$Q_MSGNAME, MAIL$Q_MSGTYPE
.EXTRN MAIL$Q_RD_DESC, MAIL$L_NMSG
.EXTRN MAIL$L_CURMSG, MAIL$L_MSGTBL
```

.PSECT \$CODE\$,NOWRT,2

04	00000000G	5E	0000	00000	.ENTRY	MAIL\$OPEN_IF, Save nothing	:	0401
		00	08	C2 00002	SUBL2	#8, SP	:	
		50	03	E1 00005	BBC	#3, MAIL\$GL_FLAGS+1, 1\$:	0421
			01	D0 0000D	MOVL	#1, R0	:	0422
				04 00010	RET		:	
			6E	7C 00011	CLRQ	DESC	:	0424
			5E	DD 00013	PUSHL	SP	:	0425
	00000000G	00	01	FB 00015	CALLS	#1, MAIL\$OPEN_FILE	:	
				04 0001C	RET		:	0426

MAIL\$CMDS
V04-000

VAX/VMS PERSONAL MAIL UTILITY COMMANDS

N 5
16-Sep-1984 00:47:25
14-Sep-1984 12:42:16

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[MAIL.SRC]CMDS.B32;1

Page 12
(3)

; Routine Size: 29 bytes, Routine Base: \$CODE\$ + 0284


```
: 285      0427 1 ROUTINE INIT_SELECT_DATA : NOVALUE =  
: 286      0428 1 ++  
: 287      0429 1 FUNCTIONAL DESCRIPTION:  
: 288      0430 1  
: 289      0431 1 Initialize the message selection data vector  
: 290      0432 1  
: 291      0433 1 INPUTS:  
: 292      0434 1  
: 293      0435 1 None  
: 294      0436 1  
: 295      0437 1 OUTPUTS:  
: 296      0438 1  
: 297      0439 1 SELECT_DATA vector initialized for parsing  
: 298      0440 1 --  
: 299      0441 2 BEGIN  
: 300      0442 2 SELECT_DATA[0] = 0;  
: 301      0443 2 SELECT_DATA[1] = -1;  
: 302      0444 2 SELECT_DATA[2] = -1;  
: 303      0445 2 SELECT_DATA[3] = 0;  
: 304      0446 2 SELECT_DATA[4] = 1;  
: 305      0447 2 RETURN  
: 306      0448 1 END;
```

```
0004 00000 INIT_SELECT_DATA:  
52 00000000' 00 9E 00002 .WORD Save R2  
62 D4 00009 MOVAB SELECT_DATA, R2  
04 A2 01 CE 0000B CLRL SELECT_DATA  
08 A2 01 CE 0000F MNEGL #1, SELECT_DATA+4  
10 A2 0C A2 D4 00013 MNEGL #1, SELECT_DATA+8  
01 D0 00016 CLRL SELECT_DATA+12  
04 0001A MOVL #1, SELECT_DATA+16  
RET
```

```
: 0427  
: 0442  
: 0443  
: 0444  
: 0445  
: 0446  
: 0448
```

; Routine Size: 27 bytes, Routine Base: \$CODE\$ + 02A1

```
0449 1 ROUTINE PARSE_SELECTION_QUALS =
0450 1 ++
0451 1 FUNCTIONAL DESCRIPTION:
0452 1
0453 1 Parse the message selection qualifiers and fill in SELECT_DATA
0454 1 vector.
0455 1
0456 1 INPUTS:
0457 1
0458 1 SELECT_DATA vector initialized for parsing
0459 1
0460 1 OUTPUTS:
0461 1
0462 1 SELECT_DATA vector filled in according to selection criteria in command
0463 1 --
0464 2 BEGIN
0465 2
0466 2 BIND
0467 2 QUAL_FLAGS = SELECT_DATA[0] : $BBLOCK;
0468 2
0469 2 Parse the /BEFORE qualifier
0470 2
0471 2 QUAL_FLAGS[QUAL_BEFORE] = CL$PRESENT(SD_BEFORE);
0472 2 IF .QUAL_FLAGS[QUAL_BEFORE]
0473 2 THEN BEGIN
0474 2 CL$GET_VALUE(SD_BEFORE,DYN_DESC);
0475 2 IF .DYN_DESC[DSC$W_LENGTH] EQL 0
0476 2 THEN STR$COPY DX(DYN_DESC,SD_TODAY);
0477 2 IF_ERR(LIB$CVT_TIME(DYN_DESC,SELECT_DATA[1]);
0478 2 RETURN SIGNAL(MAIL$_INVQUAVAL,2,DYN_DESC,SD_BEFORE));
0479 2 IF .SELECT_DATA[2] EQL 0
0480 2 OR (.SELECT_DATA[2] AND %X'FFF80000') EQL 0
0481 2 THEN RETURN SIGNAL(MAIL$_INVQUAVAL,2,DYN_DESC,SD_BEFORE);
0482 2 END;
0483 2
0484 2 Parse the /SINCE qualifier
0485 2
0486 2 QUAL_FLAGS[QUAL_SINCE] = CL$PRESENT(SD_SINCE);
0487 2 IF .QUAL_FLAGS[QUAL_SINCE]
0488 2 THEN BEGIN
0489 2 CL$GET_VALUE(SD_SINCE,DYN_DESC);
0490 2 IF .DYN_DESC[DSC$W_LENGTH] EQL 0
0491 2 THEN STR$COPY DX(DYN_DESC,SD_TODAY);
0492 2 IF_ERR(LIB$CVT_TIME(DYN_DESC,SELECT_DATA[3]);
0493 2 RETURN SIGNAL(MAIL$_INVQUAVAL,2,DYN_DESC,SD_SINCE));
0494 2 IF .SELECT_DATA[4] EQL 0
0495 2 OR (.SELECT_DATA[4] AND %X'FFF80000') EQL 0
0496 2 THEN RETURN SIGNAL(MAIL$_INVQUAVAL,2,DYN_DESC,SD_SINCE);
0497 2 END;
0498 2 !QUAL_FLAGS[QUAL_FROM] = CL$GET_VALUE(SD_FROM,SELECT_FROM);
0499 2 QUAL_FLAGS[QUAL_NEW] = CL$PRESENT(SD_NEW);
0500 2 ! QUAL_REPLIED
0501 2
0502 2 RETURN TRUE
0503 1 END;
```


QUAL_FLAGS=

SELECT_DATA

				00FC 00000	PARSE_SELECTION_QUALS:			
					.WORD	Save R2,R3,R4,R5,R6,R7	0449	
			57	00000000G	00 9E 00002	MOVAB	LIB\$CVT_TIME, R7	
			56	00000000G	00 9E 00009	MOVAB	STR\$COPY_DX, R6	
			55	00000000G	00 9E 00010	MOVAB	CLISGET_VALUE, R5	
			54	00000000G	00 9E 00017	MOVAB	CLISPRESENT, R4	
			53	FD36	CF 9E 0001E	MOVAB	SD_BEFORE, R3	
			52	00000000	00 9E J0023	MOVAB	DYN_DESC, R2	
					53 DD 0002A	PUSHL	R3	0471
			64		01 FB 0002C	CALLS	#1, CLISPRESENT	
DC	A2	01	00		50 F0 0002F	INSV	R0, #0, #1, QUAL_FLAGS	
			32	DC	A2 E9 00035	BLBC	QUAL_FLAGS, 3\$	0472
					52 DD 00039	PUSHL	R2	0474
					53 DD 0003B	PUSHL	R3	
			65		02 FB 0003D	CALLS	#2, CLISGET_VALUE	
					62 B5 00040	TSTW	DYN_DESC	0475
					09 12 00042	BNEQ	1\$	
				0178	C3 9F 00044	PUSHAB	SD_TODAY	0476
					52 DD 00048	PUSHL	R2	
			66		02 FB 0004A	CALLS	#2, STR\$COPY_DX	
				E0	A2 9F 0004D 1\$:	PUSHAB	SELECT_DATA+4	0478
					52 DD 00050	PUSHL	R2	
			67		02 FB 00052	CALLS	#2, LIB\$CVT_TIME	
			0F		50 E9 00055	BLBC	STATUS, 2\$	
			50	E4	A2 D0 00058	MOVL	SELECT_DATA+8, R0	0479
					09 13 0005C	BEQL	2\$	
			FFF80000	8F	50 D3 0005E	BITL	R0, #-524288	0480
					04 12 00065	BNEQ	3\$	
					53 DD 00067 2\$:	PUSHL	R3	0481
					46 11 00069	BRB	6\$	
				0200	C3 9F 0006B 3\$:	PUSHAB	SD_SINCE	0486
			64		01 FB 0006F	CALLS	#1, CLISPRESENT	
DC	A2	01	01		50 F0 00072	INSV	R0, #1, #1, QUAL_FLAGS	
		46	DC	A2	01 E1 00078	BBC	#1, QUAL_FLAGS, 7\$	0487
					52 DD 0007D	PUSHL	R2	0489
				0200	C3 9F 0007F	PUSHAB	SD_SINCE	
			65		02 FB 00083	CALLS	#2, CLISGET_VALUE	
					62 B5 00086	TSTW	DYN_DESC	0490
					09 12 00088	BNEQ	4\$	
				0178	C3 9F 0008A	PUSHAB	SD_TODAY	0491
					52 DD 0008E	PUSHL	R2	
			66		02 FB 00090	CALLS	#2, STR\$COPY_DX	
				E8	A2 9F 00093 4\$:	PUSHAB	SELECT_DATA+T2	0493
					52 DD 00096	PUSHL	R2	
			67		02 FB 00098	CALLS	#2, LIB\$CVT_TIME	
			0F		50 E9 0009B	BLBC	STATUS, 5\$	
			50	EC	A2 D0 0009E	MOVL	SELECT_DATA+16, R0	0494
					09 13 000A2	BEQL	5\$	
			FFF80000	8F	50 D3 000A4	BITL	R0, #-524288	0495
					16 12 000AB	BNEQ	7\$	
				0200	C3 9F 000AD 5\$:	PUSHAB	SD_SINCE	0496
					52 DD 000B1 6\$:	PUSHL	R2	
					02 DD 000B3	PUSHL	#2	

MAIL\$CMDS
V04-000

VAX/VMS PERSONAL MAIL UTILITY COMMANDS

E 6
16-Sep-1984 00:47:25
14-Sep-1984 12:42:16

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[MAIL.SRC]CMDS.B32;1

Page 16
(5)

00000000G	00	007E132A	8F	DD	000B5	PUSHL	#8262442	:
			04	FB	000BB	CALLS	#4, LIB\$SIGNAL	:
				04	000C2	RET		:
		01B0	C3	9F	000C3	PUSHAB	SD_NEW	0499
DC	A2		01	FB	000C7	CALLS	#1, CLIP\$PRESENT	:
			50	FO	000CA	INSV	R0, #4, #1, QUAL_FLAGS	:
			01	DO	000D0	MOVL	#1, R0	0502
				04	000D3	RET		0503

; Routine Size: 212 bytes, Routine Base: \$CODE\$ + 02BC


```
0504 1 ROUTINE READ_MESSAGE (EDIT_FLAG) =
0505 1 ++
0506 1 FUNCTIONAL DESCRIPTION:
0507 1
0508 1 Begin reading the next message
0509 1
0510 1 INPUTS:
0511 1
0512 1 EDIT_FLAG = TRUE if to invoke editor, false if screen-at-a-time read
0513 1 MAIL$REWIND_I_MSG or MAIL$REWIND_MSG has been called as appropriate
0514 1
0515 1 OUTPUTS:
0516 1
0517 1 Calls MAIL$READ to read and display text of message
0518 1 --
0519 2 BEGIN
0520 2
0521 2 IF .EDIT_FLAG
0522 3 THEN BEGIN
0523 3 RETURN MAIL$EDIT_READ_MSG()
0524 3 END
0525 3 ELSE BEGIN
0526 3 MAIL$B_MSGTRAN = MMB_C_FROMXDT;
0527 3 RETURN MAIL$READ()
0528 2 END;
0529 1 END;
```

!Don't put date/time on from line

```
0000 00000 READ_MESSAGE:
      08      04 AC E9 00002 .WORD Save nothing
00000000G 00      00 FB 00006 BLBC EDIT_FLAG, 1$
      04 0000D CALLS #0, MAIL$EDIT_READ_MSG
00000000G 00      06 90 0000E 1$: MOV B #6, MAIL$B_MSGTRAN
00000000G 00      00 FB 00015 CALLS #0, MAIL$READ
      04 0001C RET
```

; Routine Size: 29 bytes, Routine Base: \$CODE\$ + 0390

: 0504
: 0521
: 0523
: 0525
: 0526
: 0527
: 0529

```
: 391      0530 1 ROUTINE CHECK_FOLDER (FOLDER_DESC) =  
: 392      0531 1 ++  
: 393      0532 1 FUNCTIONAL DESCRIPTION:  
: 394      0533 1 |  
: 395      0534 1 |         Check the folder name, signal any errors.  
: 396      0535 1 |  
: 397      0536 1 |--  
: 398      0537 2 BEGIN  
: 399      0538 2  
: 400      0539 2 MAP  
: 401      0540 2     FOLDER_DESC : REF $BBLOCK;  
: 402      0541 2  
: 403      0542 2 IF .FOLDER_DESC[DSC$W_LENGTH] GTRU MFR S_FILEKEY  
: 404      0543 2     THEN RETURN SIGNAL(MAIL$_BADVALUE,T,-FOLDER_DESC)  
: 405      0544 2     ELSE RETURN TRUE  
: 406      0545 1 END;
```

				0000 00000 CHECK_FOLDER:				
	27	04	BC	B1	00002	.WORD	Save nothing	: 0530
			13	1B	00006	CMPW	@FOLDER_DESC, #39	: 0542
		04	AC	DD	00008	BLEQU	1\$	
			01	DD	0000B	PUSHL	FOLDER_DESC	: 0543
			01	DD	0000B	PUSHL	#1	
		007E1112	8F	DD	0000D	PUSHL	#8261906	
00000000G	00		03	FB	00013	CALLS	#3, LIB\$SIGNAL	
				04	0001A	RET		: 0544
	50		01	D0	0001B	MOVL	#1, R0	
			04	0001E	RET			: 0545

; Routine Size: 31 bytes, Routine Base: \$CODE\$ + 03AD


```

: 408      0546 1 ROUTINE SELECT_DEFAULT =
: 409      0547 1 |++
: 410      0548 1 | FUNCTIONAL DESCRIPTION:
: 411      0549 1 |
: 412      0550 1 |         Open the mail file (if not open) and select the default messages
: 413      0551 1 | --
: 414      0552 2 BEGIN
: 415      0553 2
: 416      0554 2 LOCAL
: 417      0555 2     FOLDER_DESC : VECTOR[2,LONG];
: 418      0556 2
: 419      0557 2     FOLDER_DESC[0] = FOLDER_DESC[1] = 0;
: 420      0558 2     IF NOT .MAIL$GL_FLAGS[MAIF_V_FILOPN]
: 421      0559 2         OR .MAIL$Q_CURFOLDER[DESC$W_LENGTH] EQL 0
: 422      0560 2     THEN BEGIN
: 423      P 0561 3         IF_ERR(MAIL$OPEN_IF());
: 424      0562 3             RETURN .STATUS;
: 425      0563 3         IF NOT .MAIL$GL_FLAGS[MAIF_V_ISAM] !All done if not isam file
: 426      0564 3             THEN RETURN TRUE;
: 427      0565 3         INIT_SELECT_DATA();
: 428      P 0566 3         IF_ERR(MAIL$SELECT_MSGS(FOLDER_DESC,SELECT_DATA));
: 429      0567 3             RETURN .STATUS;
: 430      0568 2     END;
: 431      0569 2
: 432      0570 2 RETURN TRUE
: 433      0571 1 END;
```

```

                                0000 00000 SELECT_DEFAULT:
                                .WORD  Save nothing
                                CLRQ  FOLDER_DESC
                                BBC    #3, MAIL$GL_FLAGS+1, 1$
                                TSTW   MAIL$Q_CURFOLDER
                                BNEQ   2$
                                CALLS  #0, MAIL$OPEN_IF
                                BLBC   STATUS, 3$
                                BLBC   MAIL$GL_FLAGS, 2$
                                CALLS  #0, INIT_SELECT_DATA
                                PUSHAB SELECT_DATA
                                PUSHAB FOLDER_DESC
                                CALLS  #2, MAIL$SELECT_MSGS
                                BLBC   STATUS, 3$
                                MOVL   #1, R0
                                RET
                                04 0003E 3$:

08 00000000G 00          7E 7C 00002          .WORD  Save nothing
                                03 E1 00004          CLRQ  FOLDER_DESC
                                00 B5 0000C          BBC    #3, MAIL$GL_FLAGS+1, 1$
                                27 12 00012          TSTW   MAIL$Q_CURFOLDER
                                00 FB 00014 1$:          BNEQ   2$
                                50 E9 00019          CALLS  #0, MAIL$OPEN_IF
                                00 E9 0001C          BLBC   STATUS, 3$
                                00 FB 00023          BLBC   MAIL$GL_FLAGS, 2$
                                00 9F 00028          CALLS  #0, INIT_SELECT_DATA
                                AE 9F 0002E          PUSHAB SELECT_DATA
                                02 FB 00031          PUSHAB FOLDER_DESC
                                50 E9 00038          CALLS  #2, MAIL$SELECT_MSGS
                                01 D0 0003B 2$:          BLBC   STATUS, 3$
                                04 0003E 3$:          MOVL   #1, R0
                                RET
                                0570
                                0571
```

; Routine Size: 63 bytes, Routine Base: \$CODE\$ + 03CC

MAIL\$CMDS
V04-000

VAX/VMS PERSONAL MAIL UTILITY COMMANDS

1 6
16-Sep-1984 00:47:25
14-Sep-1984 12:42:16

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[MAIL.SRC]CMDS.B32;1 Page 20
(9)

```
: 435      0572 1 GLOBAL ROUTINE MAIL$CLR_DIR_FLAG : NOVALUE =
: 436      0573 1 ++
: 437      0574 1 FUNCTIONAL DESCRIPTION:
: 438      0575 1
: 439      0576 1 This routine clears the flags that determine whether a directory
: 440      0577 1 command was in progress.
: 441      0578 1
: 442      0579 1 --
: 443      0580 2 BEGIN
: 444      0581 2
: 445      0582 2 DIR_FLAG = 0;
: 446      0583 2 DIR_MODE = 0;
: 447      0584 2 RETURN;
: 448      0585 1 END;
```

```
00000000' 00 0000 00000
00000000' 00 94 00002
00000000' 00 94 00008
04 0000E
```

```
.ENTRY MAIL$CLR_DIR_FLAG, Save nothing
CLRB DIR_FLAG
CLRB DIR_MODE
RET
```

```
: 0572
: 0582
: 0583
: 0585
```

; Routine Size: 15 bytes, Routine Base: \$CODE\$ + 040B


```
0586 1 GLOBAL ROUTINE MAIL$CMD_READ (PARSE_FLAG) =
0587 1 ++
0588 1 FUNCTIONAL DESCRIPTION:
0589 1
0590 1 This is the READ command action routine and is called to display
0591 1 the rest of the current message text, or if there is no more,
0592 1 the text of the next message, or if a specific message is specified,
0593 1 the text of the specified message.
0594 1
0595 1 INPUTS:
0596 1
0597 1 PARSE_FLAG argument present if not called from CLISDCL_PARSE
0598 1
0599 1 OUTPUTS:
0600 1
0601 1 DIR_FLAG = Zero.
0602 1
0603 1 Specified message text is displayed on SYS$OUTPUT.
0604 1 --
0605 2 BEGIN
0606 2
0607 2 BUILTIN
0608 2 ACTUALCOUNT; ! Number of actual args passed
0609 2
0610 2 LOCAL
0611 2 FOLDER_DESC : $BBLOCK[DSC$C_S_BLN],
0612 2 NEW_FILE,
0613 2 EDIT_FLAG,
0614 2 MSG_NUM;
0615 2
0616 2 MAIL$CLR_DIR_FLAG();
0617 2 MSG_NUM = 0;
0618 2 EDIT_FLAG = 0;
0619 2 INIT_SDESC(FOLDER_DESC);
0620 2 INIT_SELECT_DATA();
0621 2 IF ACTUALCOUNT() EQL 0 !If called from CLIDISPATCH
0622 3 THEN BEGIN
0623 3 IF CLISGET_VALUE(SD_READ1,DYN_DESC_3) !If first parameter present
0624 4 THEN BEGIN
0625 4
0626 4 See if it's a number. If not, it must be the foldername
0627 4
0628 4 IF NOT LIB$CVT_DTB(.DYN_DESC_3[DSC$W_LENGTH],
0629 4 .DYN_DESC_3[DSC$A_POINTER],MSG_NUM)
0630 5 THEN BEGIN
0631 5 FOLDER_DESC[DSC$W_LENGTH] = .DYN_DESC_3[DSC$W_LENGTH];
0632 5 FOLDER_DESC[DSC$A_POINTER] = .DYN_DESC_3[DSC$A_POINTER];
0633 4 END;
0634 3 END;
0635 3
0636 3 ! If 2nd parameter present it MUST be the message number
0637 3
0638 3 IF CLISGET_VALUE(SD_READ2,DYN_DESC_2)
0639 4 THEN BEGIN
0640 4 IF NOT LIB$CVT_DTB(.DYN_DESC_2[DSC$W_LENGTH],
0641 4 .DYN_DESC_2[DSC$A_POINTER],MSG_NUM)
0642 4 THEN RETURN SIGNAL(MAIL$_BADVALUE,1,DYN_DESC_2);
```

```
.. 507      0643      |      END;
508      0644      |
509      0645      |      | Parse message selection qualifiers
510      0646      |
511      P 0647      |      IF_ERR(CHECK_FOLDER(FOLDER_DESC);,
512      0648      |          RETURN .STATUS);
513      P 0649      |      IF_ERR(PARSE_SELECTION_QUALS();,
514      0650      |          RETURN .STATUS);
515      0651      |      EDIT_FLAG = CLIPRESENT(SD_EDIT);
516      0652      |      IF .SELECT_FLAGS[QUAL_NEW]
517      0653      |          AND NOT .MAIL$GL_FLAGS[MAIF_V_DFMSG]
518      0654      |          THEN MAIL$CLOSE_FILE();
519      0655      |      END;
520      0656      |
521      0657      |      |
522      0658      |      | If there are no more records in the current message or a specific message
523      0659      |      | is desired, find the next or specific message, respectively.
524      0660      |      |
525      0661      |      IF (NEW_FILE = (NOT .MAIL$GL_FLAGS[MAIF_V_FILOPN]
526      0662      |          OR (.FOLDER_DESC[DESC$W_LENGTH] NEQ 0)
527      0663      |          OR (.SELECT_DATA[0] NEQ 0)))
528      P 0664      |      THEN IF_ERR(MAIL$OPEN IF();,
529      0665      |          RETURN .STATUS);
530      0666      |      IF .MAIL$GL_FLAGS[MAIF_V_ISAM]
531      0667      |      THEN BEGIN
532      0668      |          IF .NEW_FILE
533      0669      |              OR .MAIL$L_NMSG EQL 0
534      P 0670      |              THEN (IF_ERR(MAIL$SELECT_MSGS(FOLDER_DESC,SELECT_DATA);,
535      0671      |                  RETURN .STATUS);
536      0672      |                  MAIL$GL_FLAGS[MAIF_V_RDHOLD] = 0);
537      0673      |      IF .MAIL$L_MSG_STS EQL 0
538      0674      |          OR .MAIL$L_MSG_STS EQL MAIL$_NOTREADIN
539      0675      |          OR .MSG_NUM NEQ 0
540      0676      |      THEN BEGIN
541      0677      |          IF MAIL$FIND_I_MSG(.MSG_NUM,(IF .MSG_NUM NEQ 0
542      0678      |              THEN 0
543      0679      |              ELSE 1))
544      0680      |              AND MAIL$REWIND_I_MSG()
545      0681      |          THEN
546      0682      |              READ_MESSAGE(.EDIT_FLAG)
547      0683      |          ELSE
548      0684      |              SIGNAL(.MAIL$L_MSG_STS);
549      0685      |
550      0686      |      RETURN TRUE
551      0687      |      END;
552      0688      |      |
553      0689      |      | If there aren't any more messages in the message file, start over with
554      0690      |      | the first message in the file.
555      0691      |      |
556      0692      |      IF .MAIL$L_MSG_STS EQL MAIL$_NOMOREMSG
557      0693      |      THEN BEGIN
558      0694      |          IF MAIL$FIND_I_MSG(1,1)
559      0695      |              AND MAIL$REWIND_I_MSG()
560      0696      |          THEN
561      0697      |              READ_MESSAGE(.EDIT_FLAG);
562      0698      |          IF NOT .MAIL$L_MSG_STS
563      0699      |              AND .MAIL$C_MSG_STS NEQ 0
```



```
564 0700 4 THEN SIGNAL(.MAIL$_MSG_STS);
565 0701 4 RETURN TRUE;
566 0702 3 END;
567 0703 3 END
568 0704 3 ELSE BEGIN
569 0705 3 IF .SELECT_DATA[0] NEQ 0
570 0706 3 OR .FOCDER_DESC[DSC$W_LENGTH] NEQ 0
571 0707 3 THEN RETURN SIGNAL(MAIL$_NOTISAM);
572 0708 3 IF .MAIL$_MSG_STS EQL 0
573 0709 3 OR .MAIL$_MSG_STS EQL MAIL$_NOTREADIN
574 0710 3 OR .MSG_NUM NEQ 0
575 0711 4 THEN BEGIN
576 0712 4 IF MAIL$FIND_MSG(.MSG_NUM,.NEW_FILE)
577 0713 4 THEN
578 0714 4 READ_MESSAGE(.EDIT_FLAG)
579 0715 4 ELSE
580 0716 4 IF .MAIL$_MSG_STS EQL MAIL$_NOMOREMSG
581 0717 4 THEN
582 0718 4 SIGNAL(MAIL$_NOMOREMSG);
583 0719 4 RETURN TRUE;
584 0720 3 END;
585 0721 3
586 0722 3 If there aren't any more messages in the message file, start over with
587 0723 3 the first message in the file.
588 0724 3
589 0725 3 IF .MAIL$_MSG_STS EQL MAIL$_NOMOREMSG
590 0726 4 THEN BEGIN
591 0727 4 IF MAIL$FIND_MSG(1)
592 0728 4 THEN
593 0729 4 READ_MESSAGE(.EDIT_FLAG);
594 0730 4 RETURN TRUE;
595 0731 3 END;
596 0732 2 END;
597 0733 2
598 0734 2 If there are more records in the current message, display them. Otherwise
599 0735 2 there is an error.
600 0736 2
601 0737 2 IF .MAIL$_MSG_STS
602 0738 2 THEN
603 0739 2 MAIL$READ()
604 0740 2 ELSE
605 0741 2 SIGNAL(.MAIL$_MSG_STS);
606 0742 2 RETURN TRUE
607 0743 1 END;
```

OFFC 00000

```
5B 00000000G 00 9E 00002
5A 00000000G 00 9E 00009
59 00000000G 00 9E 00010
58 00000000G 00 9E 00017
57 00000000G 00 9E 0001E
56 00000000G 00 9E 00025
```

```
.ENTRY MAIL$CMD_READ, Save R2,R3,R4,R5,R6,R7,R8,- : 0586
R9,R10,R11
MOVAB MAIL$FIND_I_MSG, R11
MOVAB LIB$CVT_DTB, R10
MOVAB CLISGET_VALUE, R9
MOVAB LIB$SIGNAL, R8
MOVAB MAIL$GL_FLAGS, R7
MOVAB MAIL$_MSG_STS, R6
```

55	FF46	CF	9E	0002C	MOVAB	READ MESSAGE, R5	...
54	00000000	00	9E	00031	MOVAB	SELECT_FLAGS, R4	...
5E		0C	C2	00038	SUBL2	#12, SP	...
7B	A5	00	FB	0003B	CALLS	#0, MAIL\$CLR_DIR_FLAG	0616
		6E	7C	0003F	CLRQ	MSG_NUM	0617
		53	D4	00041	CLRL	EDIT_FLAG	0618
	08	AE	D4	00043	CLRL	FOLDER_DESC+4	0619
FF11	C5	00	FB	00046	CALLS	#0, INIT_SELECT_DATA	0620
		6C	95	0004B	TSTB	(AP)	0621
		03	13	0004D	BEQL	1\$...
		0081	31	0004F	BRW	4\$...
	34	A4	9F	00052	PUSHAB	DYN_DESC_3	0623
	FD61	CF	9F	00055	PUSHAB	SD_READ1	...
69		02	FB	00059	CALLS	#2, CLISGET_VALUE	...
19		50	E9	0005C	BLBC	R0, 2\$...
		5E	DD	0005F	PUSHL	SP	0628
	38	A4	DD	00061	PUSHL	DYN_DESC_3+4	0629
	34	A4	3C	00064	MOVZWL	DYN_DESC_3, -(SP)	0628
7E		03	FB	00068	CALLS	#3, LIB\$CVT_DTB	...
6A		50	E8	0006B	BLBS	R0, 2\$...
0A		A4	B0	0006E	MOVW	DYN_DESC_3, FOLDER_DESC	0631
04	AE	34	A4	00073	MOVL	DYN_DESC_3+4, FOLDER_DESC+4	0632
08	AE	38	A4	00078	PUSHAB	DYN_DESC_2	0638
	2C	CF	9F	0007B	PUSHAB	SD_READ2	...
	FD4B	02	FB	0007F	CALLS	#2, CLISGET_VALUE	...
69		50	E9	00082	BLBC	R0, 3\$...
1E		5E	DD	00085	PUSHL	SP	0640
	30	A4	DD	00087	PUSHL	DYN_DESC_2+4	0641
	2C	A4	3C	0008A	MOVZWL	DYN_DESC_2, -(SP)	0640
7E		03	FB	0008E	CALLS	#3, LIB\$CVT_DTB	...
6A		50	E8	00091	BLBS	R0, 3\$...
0F		A4	9F	00094	PUSHAB	DYN_DESC_2	0642
	2C	01	DD	00097	PUSHL	#1	...
	007E1112	8F	DD	00099	PUSHL	#8261906	...
68		03	FB	0009F	CALLS	#3, LIB\$SIGNAL	...
		04	000A2	RET			...
	04	AE	9F	000A3	PUSHAB	FOLDER_DESC	0648
1D	A5	01	FB	000A6	CALLS	#1, CHECK_FOLDER	...
	6B	50	E9	000AA	BLBC	STATUS, 9\$...
FF2C	C5	00	FB	000AD	CALLS	#0, PARSE_SELECTION_QUALS	0650
	63	50	E9	000B2	BLBC	STATUS, 9\$...
		CF	9F	000B5	PUSHAB	SD_EDIT	0651
00000000G	00	01	FB	000B9	CALLS	#1, CLISPRESENT	...
	53	50	D0	000C0	MOVL	R0, EDIT_FLAG	...
0C		04	E1	000C3	BBC	#4, SELECT_FLAGS, 4\$	0652
07	01	04	E0	000C7	BBS	#4, MAIL\$GL_FLAGS+1, 4\$	0653
	00000000G	00	00	000CC	CALLS	#0, MAIL\$CLOSE_FILE	0654
		50	D4	000D3	CLRL	R0	0662
		04	AE	B5	TSTW	FOLDER_DESC	...
		02	13	000D8	BEQL	5\$...
		50	D6	000DA	INCL	R0	...
52	01	03	EF	000DC	EXTZV	#3, #1, MAIL\$GL_FLAGS+1, R2	...
		50	CA	000E2	BICL2	R0, R2	...
		50	D4	000E5	CLRL	R0	0663
		64	D5	000E7	TSTL	SELECT_DATA	...
		02	13	000E9	BEQL	6\$...
		50	D6	000EB	INCL	R0	...

FEF4	52	50	CA	000ED	6\$:	BICL2	R0, R2	0661
	52	52	D2	000F0		MCOML	R2, NEW_FILE	
	08	52	E9	000F3		BLBC	NEW_FILE, 7\$	
	C5	00	FB	000F6		CALLS	#0, MAIL\$OPEN_IF	0665
	1A	50	E9	000FB		BLBC	STATUS, 9\$	
	7D	67	E9	000FE	7\$:	BLBC	MAIL\$GL_FLAGS, 18\$	0666
	08	52	E8	00101		BLBS	NEW_FILE, 8\$	0668
		00	D5	00104		TSTL	MAIL\$GL_NMSG	0669
		13	12	0010A		BNEQ	11\$	
		54	DD	0010C	8\$:	PUSHL	R4	0671
		AE	9F	0010E		PUSHAB	FOLDER_DESC	
00000000G	00	02	FB	00111		CALLS	#2, MAIL\$SELECT_MSGS	
	01	50	E8	00118	9\$:	BLBS	STATUS, 10\$	
		04	0011B		RET			
	67	20	8A	0011C	10\$:	BICB2	#32, MAIL\$GL_FLAGS	0672
	50	66	D0	0011F	11\$:	MOVL	MAIL\$GL_MSG_STS, R0	0673
		0D	13	00122		BEQL	12\$	
007E8022	8F	50	D1	00124		CMPL	R0, #8290338	0674
		04	13	0012B		BEQL	12\$	
		6E	D5	0012D		TSTL	MSG_NUM	0675
		21	13	0012F		BEQL	16\$	
		6E	D5	00131	12\$:	TSTL	MSG_NUM	0677
		04	13	00133		BEQL	13\$	
		7E	D4	00135		CLRL	-(SP)	
		02	11	00137		BRB	14\$	
		01	DD	00139	13\$:	PUSHL	#1	
		AE	DD	0013B	14\$:	PUSHL	MSG_NUM	
	6B	02	FB	0013E		CALLS	#2, MAIL\$FIND_I_MSG	
00000000G	0A	50	E9	00141		BLBC	R0, 15\$	
	00	00	FB	00144		CALLS	#0, MAIL\$REWIND_I_MSG	0680
	61	50	E8	0014B		BLBS	R0, 22\$	
		66	DD	0014E	15\$:	PUSHL	MAIL\$GL_MSG_STS	0684
		6F	11	00150		BRB	23\$	
007E8042	8F	66	D1	00152	16\$:	CMPL	MAIL\$GL_MSG_STS, #8290370	0692
		6F	12	00159		BNEQ	25\$	
		01	DD	0015B		PUSHL	#1	0694
		01	DD	0015D		PUSHL	#1	
	6B	02	FB	0015F		CALLS	#2, MAIL\$FIND_I_MSG	
00000000G	0F	50	E9	00162		BLBC	R0, 17\$	
	00	00	FB	00165		CALLS	#0, MAIL\$REWIND_I_MSG	0695
	05	50	E9	0016C		BLBC	R0, 17\$	
		53	DD	0016F		PUSHL	EDIT_FLAG	0697
	65	01	FB	00171		CALLS	#1, READ_MESSAGE	
	50	66	D0	00174	17\$:	MOVL	MAIL\$GL_MSG_STS, R0	0698
	79	50	E8	00177		BLBS	R0, 30\$	
		77	13	0017A		BEQL	30\$	0699
		70	11	0017C		BRB	28\$	0700
		64	D5	0017E	18\$:	TSTL	SELECT_DATA	0705
		05	12	00180		BNEQ	19\$	
		AE	B5	00182		TSTW	FOLDER_DESC	0706
		0A	13	00185		BEQL	20\$	
		8F	DD	00187	19\$:	PUSHL	#8290474	0707
	68	01	FB	0018D		CALLS	#1, LIB\$SIGNAL	
		04	00190		RET			
	50	66	D0	00191	20\$:	MOVL	MAIL\$GL_MSG_STS, R0	0708
		0D	13	00194		BEQL	21\$	
007E8022	8F	50	D1	00196		CMPL	R0, #8290338	0709

		04	13	0019D	BEQL	21\$		
		6E	D5	0019F	TSTL	MSG_NUM		0710
		20	13	001A1	BEQL	24\$		
		52	DD	001A3	PUSHL	NEW_FILE		0712
		AE	DD	001A5	PUSHL	MSG_NUM		
00000000G	00	02	FB	001A8	CALLS	#2, MAIL\$FIND_MSG		
	26	50	E8	001AF	BLBS	R0, 26\$		
007E8042	8F	66	D1	001B2	CMPL	MAIL\$L_MSG_STS, #8290370		0716
		38	12	001B9	BNEQ	30\$		
		8F	DD	001BB	PUSHL	#8290370		0718
		2D	11	001C1	BRB	29\$		
007E8042	8F	66	D1	001C3	CMPL	MAIL\$L_MSG_STS, #8290370		0725
		13	12	001CA	BNEQ	27\$		
		01	DD	001CC	PUSHL	#1		0727
00000000G	00	01	FB	001CE	CALLS	#1, MAIL\$FIND_MSG		
	1B	50	E9	001D5	BLBC	R0, 30\$		
		53	DD	001D8	PUSHL	EDIT_FLAG		0729
	65	01	FB	001DA	CALLS	#1, READ_MESSAGE		
		14	11	001DD	BRB	30\$		0730
	50	66	D0	001DF	MOVL	MAIL\$L_MSG_STS, R0		0737
	09	50	E9	001E2	BLBC	R0, 28\$		
00000000G	00	00	FB	001E5	CALLS	#0, MAIL\$READ		0739
		05	11	001EC	BRB	30\$		
		50	DD	001EE	PUSHL	R0		0741
	68	01	FB	001F0	CALLS	#1, LIB\$SIGNAL		
	50	01	D0	001F3	MOVL	#1, R0		0742
		04	001F6	RET				0743

; Routine Size: 503 bytes, Routine Base: \$CODE\$ + 041A


```

: 609      0744 1 GLOBAL ROUTINE MAIL$CMD_BACK =
: 610      0745 1 ++
: 611      0746 1 FUNCTIONAL DESCRIPTION:
: 612      0747 1
: 613      0748 1 This is the BACK command action routine and is called to display
: 614      0749 1 the previous message.
: 615      0750 1
: 616      0751 1 INPUTS:
: 617      0752 1
: 618      0753 1 Current message context.
: 619      0754 1
: 620      0755 1 OUTPUTS:
: 621      0756 1
: 622      0757 1 The message before the current message is displayed on SYS$OUTPUT.
: 623      0758 1 --
: 624      0759 2 BEGIN
: 625      0760 2 LOCAL
: 626      0761 2 EDIT_FLAG,
: 627      0762 2 MSGNUM;
: 628      0763 2
: 629      0764 2 IF .DIR_MODE
: 630      0765 2 AND .MAIL$GL_FLAGS[MAIF_V_ISAM]
: 631      0766 3 THEN BEGIN
: 632      0767 3 MSGNUM = MAX(.MAIL$ MSG_NUM-2*(.MAIL$ _OUTLINES-7),1);
: 633      0768 3 IF MAIL$FIND I MSG(.MSGNUM,-1,1)
: 634      0769 3 THEN MAIL$DIRECTORY(.DIRFULL_FLAG);
: 635      0770 3 DIR_FLAG = 0;
: 636      0771 3 IF .MAIL$ MSG_STS
: 637      0772 3 THEN DIR_FLAG = 1;
: 638      0773 3 RETURN TRUE;
: 639      0774 2 END;
: 640      0775 2
: 641      0776 2 If positioned at the first message, then signal no more messages.
: 642      0777 2
: 643      0778 2 IF (.MAIL$ MSG_NUM - 1) LEQ 0
: 644      0779 3 THEN BEGIN
: 645      0780 3 IF .MAIL$GL_FLAGS[MAIF_V_ISAM]
: 646      0781 4 THEN BEGIN
: 647      0782 4 IF .MAIL$ NMSG EQL 0
: 648      0783 4 THEN MAIL$ MSG_STS = MAIL$ _NOMSGS
: 649      0784 4 ELSE MAIL$ MSG_STS = MAIL$ _NOMOREMSG;
: 650      0785 4 SIGNAL(.MAIL$ MSG_STS);
: 651      0786 4 END
: 652      0787 3 ELSE SIGNAL(MAIL$ _NOMOREMSG);
: 653      0788 3 RETURN TRUE;
: 654      0789 2 END;
: 655      0790 2 EDIT_FLAG = CLIPRESENT(SD_EDIT);
: 656      0791 2
: 657      0792 2 If positioned past the last message, then display the last message.
: 658      0793 2
: 659      0794 2 IF .MAIL$ MSG_STS EQL MAIL$ _NOMOREMSG
: 660      0795 3 THEN BEGIN
: 661      0796 3 IF .MAIL$GL_FLAGS[MAIF_V_ISAM]
: 662      0797 4 THEN BEGIN
: 663      0798 4 MAIL$FIND I MSG(.MAIL$ NMSG,-1);
: 664      0799 4 IF NOT .MAIL$ MSG_STS
: 665      0800 4 THEN (SIGNAL(.MAIL$ MSG_STS); RETURN TRUE);
```

```
666 0801 4 MAIL$REWIND_I_MSG();
667 0802 4 END
668 0803 3 ELSE MAIL$FIND MSG(.MAIL$L_MSG_NUM);
669 0804 3 READ_MESSAGE(.EDIT_FLAG);
670 0805 3 RETURN TRUE;
671 0806 2 END;
672 0807 2
673 0808 2 Otherwise, just display the previous message.
674 0809 2
675 0810 2 IF .MAIL$GL_FLAGS[MAIF_V_ISAM]
676 0811 3 THEN BEGIN
677 0812 3 IF MAIL$FIND_I_MSG(.MAIL$L_MSG_NUM - 1,-1)
678 0813 3 THEN MAIL$REWIND_I_MSG();
679 0814 3 ELSE IF NOT .MAIL$L_MSG_STS
680 0815 3 THEN RETURN SIGNAL(.MAIL$L_MSG_STS);
681 0816 3 END
682 0817 2 ELSE MAIL$FIND MSG(.MAIL$L_MSG_NUM-1);
683 0818 2 READ_MESSAGE(.EDIT_FLAG);
684 0819 2
685 0820 2 RETURN TRUE
686 0821 2
687 0822 1 END;
```

07FC 00000

```
5A 00000000G 00 9E 00002
59 FB8A CF 9E 00009
58 00000000G 00 9E 0000E
57 00000000G 00 9E 00015
56 00000000' 00 9E 0001C
55 00000000G 00 9E 00023
54 00000000G 00 9E 0002A
53 00000000G 00 9E 00031
3E 02 A6 E9 00038
3B 64 E9 0003C
51 65 D0 0003F
50 00000000G 00 D0 00042
50 02 C4 00049
51 50 C2 0004C
50 0E A1 9E 0004F
03 14 00053
50 01 D0 00055
01 DD 00058 1$:
7E 01 CE 0005A
50 DD 0005D
67 03 FB 0005F
0B 50 E9 00062
7E 01 A6 9A 00065
00000000G 00 01 FB 00069
66 66 94 00070 2$:
58 63 E9 00072
66 01 90 00075
53 11 00078
```

```
.ENTRY MAIL$CMD_BACK, Save R2,R3,R4,R5,R6,R7,R8,- 0744
R9,R10
MOVAB LIB$SIGNAL, R10
MOVAB SD EDIT, R9
MOVAB MAIL$L_NMSG, R8
MOVAB MAIL$FIND_I_MSG, R7
MOVAB DIR_FLAG, R6
MOVAB MAIL$L_MSG_NUM, R5
MOVAB MAIL$GL_FLAGS, R4
MOVAB MAIL$L_MSG_STS, R3
BLBC DIR_MODE, 3$ 0764
BLBC MAIL$GL_FLAGS, 3$ 0765
MOVL MAIL$L_MSG_NUM, R1 0767
MOVL MAIL$L_OUTLINES, R0
MULL2 #2, R0
SUBL2 R0, R1
MOVAB 14(R1), R0
BGTR 1$
MOVL #1, R0
PUSHL #1 0768
MNEGL #1, -(SP)
PUSHL MSGNUM
CALLS #3, MAIL$FIND_I_MSG
BLBC R0, 2$
MOVZBL DIRFULL_FLAG, -(SP) 0769
CALLS #1, MAIL$DIRECTORY
CLRB DIR_FLAG 0770
BLBC MAIL$L_MSG_STS, 9$ 0771
MOVB #1, DIR_FLAG 0772
BRB 9$ 0773
```


01	65	D1	0007A	3\$:	CMPL	MAIL\$ _MSG_NUM, #1	: 0778
	23	14	0007D		BGTR	7\$: 0780
18	64	E9	0007F		BLBC	MAIL\$GL_FLAGS, 6\$: 0782
	68	D5	00082		TSTL	MAIL\$ _NMSG	: 0783
	09	12	00084		BNEQ	4\$: 0784
63 007E804A	8F	D0	00086		MOVL	#8290378, MAIL\$ _MSG_STS	: 0785
	07	11	0008D		BRB	5\$: 0787
63 007E8042	8F	D0	0008F	4\$:	MOVL	#8290370, MAIL\$ _MSG_STS	: 0790
	63	DD	00096	5\$:	PUSHL	MAIL\$ _MSG_STS	: 0794
	30	11	00098		BRB	8\$: 0796
007E8042	8F	DD	0009A	6\$:	PUSHL	#8290370	: 0798
	28	11	000A0		BRB	8\$: 0799
	59	DD	000A2	7\$:	PUSHL	R9	: 0800
00000000G 00	01	FB	000A4		CALLS	#1, CLIS\$PRESENT	: 0803
	52	50	D0	000AB	MOVL	R0, EDIT_FLAG	: 0812
007E8042 8F	63	D1	000AE		CMPL	MAIL\$ _MSG_STS, #8290370	: 0810
	1C	12	000B5		BNEQ	11\$: 0812
	64	E9	000B7		BLBC	MAIL\$GL_FLAGS, 10\$: 0813
	01	CE	000BA		MNEGL	#1, -(SP)	: 0814
	68	DD	000BD		PUSHL	MAIL\$ _NMSG	: 0815
	02	FB	000BF		CALLS	#2, MAIL\$FIND_I_MSG	: 0817
67	63	D0	000C2		MOVL	MAIL\$ _MSG_STS, R0	: 0818
50	50	E8	000C5		BLBS	R0, 12\$: 0820
1D	50	DD	000C8		PUSHL	R0	: 0822
	01	FB	000CA	8\$:	CALLS	#1, LIB\$SIGNAL	: 0822
6A	3B	11	000CD	9\$:	BRB	17\$: 0822
	65	DD	000CF	10\$:	PUSHL	MAIL\$ _MSG_NUM	: 0822
	29	11	000D1		BRB	15\$: 0822
50	01	C3	000D3	11\$:	SUBL3	#1, MAIL\$ _MSG_NUM, R0	: 0822
	64	E9	000D7		BLBC	MAIL\$GL_FLAGS, 14\$: 0822
	01	CE	000DA		MNEGL	#1, -(SP)	: 0822
	50	DD	000DD		PUSHL	R0	: 0822
	02	FB	000DF		CALLS	#2, MAIL\$FIND_I_MSG	: 0822
	50	E9	000E2		BLBC	R0, 13\$: 0822
00000000G 00	00	FB	000E5	12\$:	CALLS	#0, MAIL\$REWIND_I_MSG	: 0822
	15	11	000EC		BRB	16\$: 0822
	50	D0	000EE	13\$:	MOVL	MAIL\$ _MSG_STS, R0	: 0822
	0F	50	E8	000F1	BLBS	R0, 16\$: 0822
	50	DD	000F4		PUSHL	R0	: 0822
6A	01	FB	000F6		CALLS	#1, LIB\$SIGNAL	: 0822
		04	000F9		RET		: 0822
	50	DD	000FA	14\$:	PUSHL	R0	: 0822
00000000G 00	01	FB	000FC	15\$:	CALLS	#1, MAIL\$FIND_MSG	: 0822
	52	DD	00103	16\$:	PUSHL	EDIT_FLAG	: 0822
FC75 CF	01	FB	00105		CALLS	#1, READ_MESSAGE	: 0822
50	01	D0	0010A	17\$:	MOVL	#1, R0	: 0822
	04	0010D			RET		: 0822

; Routine Size: 270 bytes, Routine Base: \$CODE\$ + 0611

```

: 689      0823 1 GLOBAL ROUTINE MAIL$CMD_CONTINUE =
: 690      0824 1 ++
: 691      0825 1 FUNCTIONAL DESCRIPTION:
: 692      0826 1
: 693      0827 1     This is the <RETURN> command action routine and is called to continue
: 694      0828 1     the last display type command.  If the last command was a READ command,
: 695      0829 1     then the READ command action routine will be called.  If the last
: 696      0830 1     command was a DIRECTORY command and there are more messages to be
: 697      0831 1     displayed, then the DIRECTORY routine will be called.
: 698      0832 1
: 699      0833 1 INPUTS:
: 700      0834 1
: 701      0835 1     DIR_FLAG = 1 if last command was a DIRECTORY command and there are
: 702      0836 1     more messages to be displayed.
: 703      0837 1
: 704      0838 1 OUTPUTS:
: 705      0839 1
: 706      0840 1     Appropriate action routine selected.
: 707      0841 1 --
: 708      0842 2 BEGIN
: 709      0843 2 IF .DIR_FLAG                                ! If more to display
: 710      0844 3 THEN BEGIN
: 711      0845 3     MAIL$DIRECTORY(.DIRFULL_FLAG);
: 712      0846 3     IF NOT .MAIL$MSG_STS                    ! If no more messages
: 713      0847 3     THEN
: 714      0848 3         DIR_FLAG = 0;                        ! then clear flag
: 715      0849 3     END
: 716      0850 2 ELSE
: 717      0851 2     MAIL$CMD_READ(1);
: 718      0852 2 RETURN TRUE
: 719      0853 1 END;
```

		0004 00000	.ENTRY MAIL\$CMD_CONTINUE, Save R2	: 0823
	52 00000000'	00 9E 00002	MOVAB DIR_FLAG, R2	: 0843
	16	62 E9 00009	BLBC DIR_FLAG, 1\$: 0845
	7E 01	A2 9A 0000C	MOVZBL DIRFULL_FLAG, -(SP)	: 0846
00000000G	00	01 FB 00010	CALLS #1, MAIL\$DIRECTORY	: 0848
	0B 00000000G	00 E8 00017	BLBS MAIL\$MSG_STS, 2\$: 0843
		62 94 0001E	CLRB DIR_FLAG	: 0851
		07 11 00020	BRB 2\$: 0852
		01 DD 00022 1\$:	PUSHL #1	: 0853
FCD2	CF	01 FB 00024	CALLS #1, MAIL\$CMD_READ	
	50	01 D0 00029 2\$:	MOVL #1, R0	
		04 0002C	RET	

; Routine Size: 45 bytes, Routine Base: \$CODE\$ + 071F

; 720 0854 1

MAIL\$CMDS
V04-000

VAX/VMS PERSONAL MAIL UTILITY COMMANDS

G 7
16-Sep-1984 00:47:25
14-Sep-1984 12:42:16

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[MAIL.SRC]CMDS.B32;1

Page 31
(13)

```
: 722      0855 1 ROUTINE TELL_CREATED(DESC) =  
: 723      0856 1 ++  
: 724      0857 1 FUNCTIONAL DESCRIPTION:  
: 725      0858 1  
: 726      0859 1 Signal that an ISAM file was created for the copy command  
: 727      0860 1  
: 728      0861 1 --  
: 729      0862 1 SIGNAL(MAIL$_CREATED,1,.DESC);
```

```
0000 00000 TELL_CREATED:  
04 AC DD 00002 .WORD Save nothing  
01 DD 00005 PUSHL DESC  
8F DD 00007 PUSHL #1  
03 FB 0000D PUSHL #8261745  
04 00014 CALLS #3, LIB$SIGNAL  
RET
```

```
: 0855  
: 0862  
:  
:  
:  
:  
:
```

; Routine Size: 21 bytes, Routine Base: \$CODE\$ + 074C

```
0863 1 GLOBAL ROUTINE MAIL$CMD_COPY =
0864 1 ++
0865 1 FUNCTIONAL DESCRIPTION:
0866 1
0867 1 This is the COPY command, which takes a copy of the current message
0868 1 and copies it to the specified folder/file
0869 1
0870 1 --
0871 2 BEGIN
0872 2 LOCAL
0873 2 COPY_FLAGS : $BBLOCK[4];
0874 2
0875 2 CLISGET_VALUE(SD_FOLDER_NAME,DYN_DESC);
P 0876 2 IF_ERR(CHECK_FOLDER(DYN_DESC),,
0877 2 RETURN .STATUS);
0878 2
0879 2 COPY_FLAGS = 0;
0880 2 COPY_FLAGS[MCR_V_ALL] = CLISPRESENT(SD_ALL) AND 1;
0881 2
0882 2 CLISGET_VALUE(SD_FILE,DYN_DESC_2);
0883 2
0884 2 IF NOT .COPY_FLAGS[MCR_V_ALL]
0885 2 AND (IF .MAIL$GL_FLAGS[MAIF_V_ISAM]
0886 2 THEN .MAIL$SL_CURMSG_EQL 0
0887 2 ELSE NOT MAIL$REWIND MSG())
0888 2 THEN RETURN SIGNAL(MAIL$_NOTREADIN);
0889 2
0890 2 IF CLISPRESENT(SD_CONFIRM)
0891 2 THEN COPY_FLAGS[MCR_V_CONFNEW] = 1;
0892 2
0893 2 MAIL$COPY_MSGS(DYN_DESC_2,DYN_DESC,.COPY_FLAGS,TELL_CREATED);
0894 2 RETURN TRUE
0895 1 END;
```

				007C 00000	.ENTRY MAIL\$CMD_COPY, Save R2,R3,R4,R5,R6	0863
		56	00000000G	00 9E 00002	MOVAB CLISPRESENT, R6	
		55	00000000G	00 9E 00009	MOVAB CLISGET_VALUE, R5	
		54	000000000	00 9E 00010	MOVAB DYN_DESC, R4	
		53	F914	CF 9E 00017	MOVAB SD_FOLDER_NAME, R3	
				18 BB 0001C	PUSHR #1, R3, R4, R5	0875
		65		02 FB 0001E	CALLS #2, CLISGET_VALUE	
				54 DD 00021	PUSHL R4	0877
	FC24	CF		01 FB 00023	CALLS #1, CHECK_FOLDER	
		67		50 E9 00028	BLBC STATUS, 5\$	
				52 D4 0002B	CLRL COPY_FLAGS	0879
			FF74	C3 9F 0002D	PUSHAB SD_ALL	0880
		66		01 FB 00031	CALLS #1, CLISPRESENT	
52	01	01		50 F0 00034	INSV R0, #1, #1, COPY_FLAGS	
			08	A4 9F 00039	PUSHAB DYN_DESC_2	0882
			EC	A3 9F 0003C	PUSHAB SD_FILE	
		65		02 FB 0003F	CALLS #2, CLISGET_VALUE	
	29	52		01 E0 00042	BBS #1, COPY_FLAGS, 3\$	0884
		0A	00000000G	00 E9 00046	BLBC MAIL\$GL_FLAGS, 1\$	0885

MAIL\$CMDS
V04-000

VAX/VMS PERSONAL MAIL UTILITY COMMANDS

I 7
16-Sep-1984 00:47:25
14-Sep-1984 12:42:16

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[MAIL.SRC]CMDS.B32;1

Page 33
(14)

		00000000G	00	D5	0004D	TSTL	MAIL\$ _CURMSG	:	0886
			1A	12	00053	BNEQ	3\$:	
			0A	11	00055	BRB	2\$:	
00000000G	00		00	FB	00057	1\$: CALLS	#0, MAIL\$REWIND_MSG	:	0887
	0E		50	E8	0005E	BLBS	R0, 3\$:	
00000000G	00	007E8022	8F	DD	00061	2\$: PUSHL	#8290338	:	0888
			01	FB	00067	CALLS	#1, LIB\$ SIGNAL	:	
				04	0006E	RET		:	
		94	A3	9F	0006F	3\$: PUSHAB	SD_CONFIRM	:	0890
	66		01	FB	00072	CALLS	#1, CLISPRESENT	:	
	05		50	E9	00075	BLBC	R0, 4\$:	
	52	0100	8F	A8	00078	BISW2	#256, COPY_FLAGS	:	0891
		FF6A	CF	9F	0007D	4\$: PUSHAB	TELL_CREATED	:	0893
			52	DD	00081	PUSHL	COPY_FLAGS	:	
			54	DD	00083	PUSHL	R4	:	
		08	A4	9F	00085	PUSHAB	DYN_DESC_2	:	
00000000G	00		04	FB	00088	CALLS	#4, MAIL\$COPY_MSGS	:	
	50		01	D0	0008F	MOVL	#1, R0	:	0894
			04	00092	5\$: RET			:	0895

; Routine Size: 147 bytes, Routine Base: \$CODE\$ + 0761

```
: 765      0896 1 GLOBAL ROUTINE MAIL$CMD_DELETE =
: 766      0897 1 ++
: 767      0898 1 FUNCTIONAL DESCRIPTION:
: 768      0899 1
: 769      0900 1     This is the DELETE command action routine and is called to delete
: 770      0901 1     the current message.
: 771      0902 1
: 772      0903 1 INPUTS:
: 773      0904 1
: 774      0905 1     Current message context.
: 775      0906 1
: 776      0907 1 OUTPUTS:
: 777      0908 1
: 778      0909 1     The current message is deleted from the file it is in.
: 779      0910 1 --
: 780      0911 2 BEGIN
: 781      0912 2 LOCAL
: 782      0913 2     DELMSGNUM,
: 783      0914 2     SAVE_MSGNUM,
: 784      0915 2     ALL_FLAG,
: 785      0916 2     STATUS;
: 786      0917 2                                     ! Routine status
: 787      0918 2 ALL_FLAG = CL$PRESENT(SD_ALL);
: 788      0919 2 IF .MAIL$GL_FLAGS[MAIF_V_ISAM]
: 789      0920 3 THEN BEGIN
: 790      0921 3     IF CL$GET_VALUE(SD_DELNUM,DYN_DESC)
: 791      0922 4     THEN BEGIN
: 792      P 0923 4         IF _ERR(LIB$CVT_DTB(.DYN_DESC[DSC$W_LENGTH],.DYN_DESC[DSC$A_POINTER],
: 793      P 0924 4             DELMSGNUM);
: 794      0925 4             RETURN SIGNAL(MAIL$BADVALUE,1,DYN_DESC));
: 795      0926 4     IF .DELMSGNUM GTRU .MAIL$L_NMSG
: 796      0927 4     THEN RETURN SIGNAL(MAIL$NOMSGS);
: 797      0928 4     IF .MAIL$L_MSGTBL[.DELMSGNUM-1] EQL 0
: 798      0929 4     THEN RETURN SIGNAL(MAIL$DELMMSG);
: 799      0930 4
: 800      0931 4     Make sure we don't point past the end of the folder
: 801      0932 4
: 802      0933 4     IF .MAIL$L_MSG_NUM GTRU .MAIL$L_NMSG THEN SAVE_MSGNUM = 0
: 803      0934 4     ELSE
: 804      0935 4         SAVE_MSGNUM = .MAIL$L_MSG_NUM;
: 805      0936 4     MAIL$L_MSG_NUM = .DELMSGNUM;
: 806      0937 4     MAIL$L_CURMSG = .MAIL$L_MSGTBL[.DELMSGNUM-1];
: 807      0938 4     STATUS=MAIL$DELETE_1_MSG();
: 808      0939 4     IF .STATUS EQL MAIL$NOTREADIN
: 809      0940 4         OR .STATUS EQL MAIL$DELWASTE
: 810      0941 4         THEN SIGNAL(.STATUS);
: 811      0942 4     MAIL$L_MSG_NUM = .SAVE_MSGNUM;
: 812      0943 4     IF .SAVE_MSGNUM NEQ 0
: 813      0944 4         THEN MAIL$L_CURMSG = .MAIL$L_MSGTBL[.SAVE_MSGNUM-1];
: 814      0945 4     RETURN TRUE
: 815      0946 3     END;
: 816      0947 3
: 817      0948 3     Delete all the messages if /ALL
: 818      0949 3
: 819      0950 3     IF .ALL_FLAG
: 820      0951 4     THEN BEGIN
: 821      0952 4         IF CH$EQL(.MAIL$Q_CURFOLDER[DSC$W_LENGTH],
```



```

: 822      0953 4      .MAIL$Q_CURFOLDER[DSC$A_POINTER],
: 823      0954 4      .MAIL$Q_WASTEDESC[DSC$W_LENGTH],
: 824      0955 4      .MAIL$Q_WASTEDESC[DSC$A_POINTER],0)
: 825      0956 4      THEN RETURN SIGNAL(MAIL$ DELWASTE);
: 826      0957 4      SAVE MSGNUM = .MAIL$ MSG_NUM;
: 827      0958 4      IF .MAIL$ NMSG GTRU 0
: 828      0959 4      THEN INCRU I FROM 1 TO .MAIL$ NMSG
: 829      0960 5      DO BEGIN
: 830      0961 5          MAIL$ CURMSG = .MAIL$ MSGTBL[I-1];
: 831      0962 5          IF .MAIL$ CURMSG NEQ 0
: 832      0963 6          THEN BEGIN
: 833      0964 6              MAIL$ MSG NUM = .I;
: 834      0965 6              MAIL$DELETE_I_MSG();
: 835      0966 5          END;
: 836      0967 4          END;
: 837      0968 4          MAIL$ MSG NUM = .SAVE_MSGNUM;
: 838      0969 4          IF .SAVE_MSGNUM NEQ 0
: 839      0970 4          THEN MAIL$ CURMSG = .MAIL$ MSGTBL[.SAVE_MSGNUM-1];
: 840      0971 4          RETURN TRUE
: 841      0972 3      END;
: 842      0973 3      !
: 843      0974 3      ! Just delete a single message
: 844      0975 3      !
: 845      0976 3      STATUS = MAIL$DELETE_I_MSG();
: 846      0977 3      IF .STATUS EQL MAIL$ NOTREADIN
: 847      0978 3      OR .STATUS EQL MAIL$ DELWASTE
: 848      0979 3      THEN SIGNAL(.STATUS)
: 849      0980 3      ELSE IF .STATUS THEN MAIL$CMD_ERASE();
: 850      0981 3      END
: 851      0982 3      ELSE BEGIN
: 852      0983 3          IF .ALL FLAG
: 853      0984 3          THEN RETURN SIGNAL(MAIL$ NOTISAM);
: 854      0985 3          STATUS = MAIL$DELETE_MSG();
: 855      0986 3          IF NOT .STATUS
: 856      0987 3          THEN
: 857      0988 3              SIGNAL(.STATUS)
: 858      0989 3          ELSE
: 859      0990 3              MAIL$CMD_ERASE();
: 860      0991 2      END;
: 861      0992 2      RETURN TRUE
: 862      0993 1      END;
```

```

! Delete the message
! Signal the failure
! Erase the screen
```

```

                                OFFC 00000
5B 00000000G 00 9E 00002
5A 00000000G 00 9E 00009
59 00000000G 00 9E 00010
58 00000000G 00 9E 00017
57 00000000' 00 9E 0001E
56 00000000G 00 9E 00025
5E          04 C2 0002C
          F7DD CF 9F 0002F
00000000G 00 01 FB 00033
```

```

.ENTRY MAIL$CMD DELETE, Save R2,R3,R4,R5,R6,R7,R8,-; 0896
R9,R10,RT1
MOVAB MAIL$ CURMSG, R11
MOVAB MAIL$ NMSG, R10
MOVAB MAIL$ MSGTBL, R9
MOVAB LIB$SIGNAL, R8
MOVAB DYN DESC, R7
MOVAB MAIL$ MSG_NUM, R6
SUBL2 #4, SP
PUSHAB SD_ALL
CALLS #1, CLISPRESNT
                                0918
```

53	03	00000000G	50	DO	0003A	MOVL	R0, ALL_FLAG	:	
			00	EB	0003D	BLBS	MAIL\$GL_FLAGS, 1\$:	0919
			0117	31	00044	BRW	20\$:	
			57	DD	00047	PUSHL	R7	:	0921
		F803	CF	9F	00049	PUSHAB	SD_DELNUM	:	
00000000G	00		02	FB	0004D	CALLS	#2, CLISGET_VALUE	:	
	03		50	EB	00054	BLBS	R0, 2\$:	
			0083	31	00057	BRW	10\$:	
			5E	DD	0005A	PUSHL	SP	:	0925
		04	A7	DD	0005C	PUSHL	DYN_DESC+4	:	
00000000G	7E		67	3C	0005F	MOVZWL	DYN_DESC, -(SP)	:	
	00		03	FB	00062	CALLS	#3, LIB\$CVT_DTB	:	
	0E		50	EB	00069	BLBS	STATUS, 3\$:	
			57	DD	0006C	PUSHL	R7	:	
			01	DD	0006E	PUSHL	#1	:	
		007E1112	8F	DD	00070	PUSHL	#8261906	:	
68			03	FB	00076	CALLS	#3, LIB\$SIGNAL	:	
			04		00079	RET		:	
6A			6E	D1	0007A	CMPL	DELMSGNUM, MAIL\$L_NMSG	:	0926
			08	1B	0007D	BLEQU	4\$:	
		007E804A	8F	DD	0007F	PUSHL	#8290378	:	0927
			7D	11	00085	BRB	11\$:	
50			69	DO	00087	MOVL	MAIL\$L_MSGTBL, R0	:	0928
52			6E	DO	0008A	MOVL	DELMSGNUM, R2	:	
		FC A042	D5	0008D	TSTL	-4(R0)[R2]		:	
			08	12	00091	BNEQ	5\$:	
		007E80B2	8F	DD	00093	PUSHL	#8290482	:	0929
			69	11	00099	BRB	11\$:	
50			66	DO	0009B	MOVL	MAIL\$L_MSG_NUM, R0	:	0933
6A			50	D1	0009E	CMPL	R0, MAIL\$L_NMSG	:	
			04	1B	000A1	BLEQU	6\$:	
			54	D4	000A3	CLRL	SAVE_MSGNUM	:	
			03	11	000A5	BRB	7\$:	
54			50	DO	000A7	MOVL	R0, SAVE_MSGNUM	:	0935
66			6E	DO	000AA	MOVL	DELMSGNUM, MAIL\$L_MSG_NUM	:	0936
50			69	DO	000AD	MOVL	MAIL\$L_MSGTBL, R0	:	0937
6B		FC A042	DO	000B0	MOVL	-4(R0)[R2], MAIL\$L_CURMSG		:	
00000000G	00		00	FB	000B5	CALLS	#0, MAIL\$DELETE_I_MSG	:	0938
	55		50	DO	000BC	MOVL	R0, STATUS	:	
007E8022	8F		55	D1	000BF	CMPL	STATUS, #8290338	:	0939
			09	13	000C6	BEQL	8\$:	
007E80BA	8F		55	D1	000C8	CMPL	STATUS, #8290490	:	0940
			05	12	000CF	BNEQ	9\$:	
			55	DD	000D1	PUSHL	STATUS	:	0941
68			01	FB	000D3	CALLS	#1, LIB\$SIGNAL	:	
66			54	DO	000D6	MOVL	SAVE_MSGNUM, MAIL\$L_MSG_NUM	:	0942
			58	12	000D9	BNEQ	17\$:	0943
			5E	11	000DB	BRB	18\$:	0945
5D			53	E9	000DD	BLBC	ALL_FLAG, 19\$:	0950
51	00000000G		00	DO	000E0	MOVL	MAIL\$Q_CURFOLDER+4, R1	:	0953
50	00000000G		00	DO	000E7	MOVL	MAIL\$Q_WASTEDESC+4, R0	:	0955
61	00000000G		00	2D	000EE	CMPC5	MAIL\$Q_CURFOLDER, (R1), #0, -	:	0952
			60		000FB		MAIL\$Q_WASTEDESC, (R0)	:	
			08	12	000FC	BNEQ	12\$:	
		007E80BA	8F	DD	000FE	PUSHL	#8290490	:	0956
			61	11	00104	BRB	21\$:	
54			66	DO	00106	MOVL	MAIL\$L_MSG_NUM, SAVE_MSGNUM	:	0957

00000000G 00

00

53	6A	D0	00109	MOVL	MAIL\$L_NMSG, R3	: 0958
	20	13	0010C	BEQL	16\$: 0959
52	01	D0	0010E	MOVL	#1, I	: 0961
	16	11	00111	BRB	15\$: 0962
50	69	D0	00113	13\$: MOVL	MAIL\$L_MSGTBL, R0	: 0964
6B	FC A042	D0	00116	MOVL	-4(R0)[I], MAIL\$L_CURMSG	: 0965
	0A	13	0011B	BEQL	14\$: 0959
66	52	D0	0011D	MOVL	I, MAIL\$L_MSG_NUM	: 0968
00000000G	00	FB	00120	CALLS	#0, MAIL\$DELETE_I_MSG	: 0969
	52	D6	00127	14\$: INCL	I	: 0970
53	52	D1	00129	15\$: CMPL	I, R3	: 0971
	E5	1B	0012C	BLEQU	13\$: 0976
66	54	D0	0012E	16\$: MOVL	SAVE_MSGNUM, MAIL\$L_MSG_NUM	: 0977
	53	13	00131	BEQL	25\$: 0978
50	69	D0	00133	17\$: MOVL	MAIL\$L_MSGTBL, R0	: 0980
6B	FC A044	D0	00136	MOVL	-4(R0)[SAVE_MSGNUM], MAIL\$L_CURMSG	: 0983
	49	11	0013B	18\$: BRB	25\$: 0984
00000000G	00	FB	0013D	19\$: CALLS	#0, MAIL\$DELETE_I_MSG	: 0985
	55	D0	00144	MOVL	R0, STATUS	: 0986
007E8022	8F	55	D1 00147	CMPL	STATUS, #8290338	: 0988
	28	13	0014E	BEQL	23\$: 0990
007E80BA	8F	55	D1 00150	CMPL	STATUS, #8290490	: 0992
	1F	13	00157	BEQL	23\$: 0993
2A	55	E9	00159	BLBC	STATUS, 25\$: 0994
	21	11	0015C	BRB	24\$: 0995
0A	53	E9	0015E	20\$: BLBC	ALL FLAG, 22\$: 0996
	8F	DD	00161	PUSHL	#8290474	: 0997
68	01	FB	00167	21\$: CALLS	#1, LIB\$SIGNAL	: 0998
	04	0016A		RET		: 0999
00000000G	00	FB	0016B	22\$: CALLS	#0, MAIL\$DELETE_MSG	: 0999
	55	D0	00172	MOVL	R0, STATUS	: 0999
07	55	E8	00175	BLBS	STATUS, 24\$: 0999
	55	DD	00178	23\$: PUSHL	STATUS	: 0999
68	01	FB	0017A	CALLS	#1, LIB\$SIGNAL	: 0999
	07	11	0017D	BRB	25\$: 0999
00000000G	00	FB	0017F	24\$: CALLS	#0, MAIL\$CMD_ERASE	: 0999
	50	D0	00186	25\$: MOVL	#1, R0	: 0999
	04	00189		RET		: 0999

; Routine Size: 394 bytes, Routine Base: \$CODE\$ + 07F4

```
.. 864      0994 1 GLOBAL ROUTINE MAIL$CMD_DIRECTORY =
865      0995 1 ++
866      0996 1 FUNCTIONAL DESCRIPTION:
867      0997 1
868      0998 1 This is the DIRECTORY command action routine and is called to
869      0999 1 display a summary of all the messages in the current file or a
870      1000 1 specified file.
871      1001 1
872      1002 1 INPUTS:
873      1003 1
874      1004 1 OUTPUTS:
875      1005 1
876      1006 1 DIR_FLAG = 1 if more messages to display.
877      1007 1
878      1008 1 A summary of each message is displayed on SYS$OUTPUT.
879      1009 1 --
880      1010 2 BEGIN
881      1011 2
882      1012 2 LOCAL
883      1013 2 FOLDER_DESC : $BBLOCK[DSC$_S_BLN],
884      1014 2 START_MSGNUM,
885      1015 2 START_FLAG,
886      1016 2 STATUS,
887      1017 2 NEW_FILE;
888      1018 2
889      1019 2 DIR_FLAG = 0;
890      1020 2 DIRFULL_FLAG = 0;
891      1021 2 INIT_SDESC(FOLDER_DESC);
892      1022 2 IF CL$GET_VALUE(SD_FOLDER_NAME,DYN_DESC_3)
893      1023 3 THEN BEGIN
894      1024 3 FOLDER_DESC[DSC$_LENGTH] = .DYN_DESC_3[DSC$_LENGTH];
895      1025 3 FOLDER_DESC[DSC$_POINTER] = .DYN_DESC_3[DSC$_POINTER];
896      P 1026 3 IF_ERR(CHECK_FOLDER(FOLDER_DESC));
897      1027 3 RETURN .STATUS;
898      1028 2 END;
899      1029 2 INIT_SELECT_DATA();
900      P 1030 2 IF_ERR(PARSE_SELECTION_QUALS());
901      1031 2 RETURN .STATUS;
902      1032 2
903      1033 2 IF .SELECT_FLAGS[QUAL_NEW]
904      1034 2 AND NOT .MAIL$GL_FLAGS[MAIF_V_DFMSG]
905      1035 2 THEN MAIL$CLOSE_FILE();
906      1036 2
907      1037 2 START_FLAG = CL$GET_VALUE(SD_START,DYN_DESC_2);
908      1038 2 DIRFULL_FLAG = CL$PRESENT(SD_FULL);
909      1039 2 IF CL$PRESENT(SD_FOLDER)
910      1040 3 THEN BEGIN
911      P 1041 3 IF_ERR(MAIL$OPEN IF());
912      1042 3 RETURN .STATUS;
913      1043 3 IF NOT .MAIL$GL_FLAGS[MAIF_V_ISAM]
914      1044 3 THEN RETURN SIGNAL(MAIL$_NOTISAM)
915      1045 3 ELSE RETURN MAIL$DIR_FOLDER(DYN_DESC_2);
916      1046 2 END;
917      1047 2
918      1048 2 IF .START_FLAG
919      1049 3 THEN BEGIN
920      P 1050 3 IF_ERR(LIB$CVT_DTB(.DYN_DESC_2[DSC$_LENGTH],.DYN_DESC_2[DSC$_POINTER],
```



```
: 921      P 1051      3      START MSGNUM);
: 922      1052      3      RETURN SIGNAL(MAIL$_BADVALUE,1,DYN_DESC_2));
: 923      1053      3      IF .START MSGNUM EQL 0
: 924      1054      3      THEN START_MSGNUM = 1;
: 925      1055      3      END
: 926      1056      2      ELSE START_MSGNUM = 0;
: 927      1057      2
: 928      1058      3      IF (NEW_FILE = NOT .MAIL$GL_FLAGS[MAIF_V_FILOPN]
: 929      1059      4      OR (.FOLDER_DESC[DSC$W_LENGTH] NEQ 0)
: 930      1060      3      OR (.SELECT_DATA[0] NEQ 0))
: 931      P 1061      2      THEN IF_ERR(MAIL$OPEN IF());
: 932      1062      2      RETURN .STATUS);
: 933      1063      2      IF .MAIL$GL_FLAGS[MAIF_V_ISAM]
: 934      1064      3      THEN BEGIN
: 935      1065      3      IF .NEW_FILE
: 936      1066      3      OR .MAIL$NMSG EQL 0
: 937      P 1067      3      THEN IF_ERR(MAIL$SELECT_MSGS(FOLDER_DESC,SELECT_DATA);,
: 938      1068      3      RETURN .STATUS);
: 939      1069      3      IF .MAIL$NMSG EQL 0
: 940      1070      3      THEN RETURN SIGNAL(MAIL$_NOMSGS);
: 941      1071      3      IF .START MSGNUM EQL 0
: 942      1072      4      THEN START_MSGNUM = (IF .NEW_FILE
: 943      1073      5      OR (.MAIL$NMSG_NUM LEQ 1)
: 944      1074      4      THEN 1
: 945      1075      4      ELSE MAX(1,MIN(.MAIL$NMSG_NUM,
: 946      1076      4      .MAIL$NMSG-.MAIL$OUTLINES+7+1)))
: 947      1077      3      ELSE START_MSGNUM = MAX(1,MIN(.START MSGNUM,
: 948      1078      3      .MAIL$NMSG-.MAIL$OUTLINES+7+1));
: 949      1079      3
: 950      1080      3      IF MAIL$FIND_I_MSG(.START MSGNUM,1,1)
: 951      1081      3      THEN MAIL$DIRECTORY(.DIRFULL_FLAG);
: 952      1082      3      END
: 953      1083      3      ELSE BEGIN
: 954      1084      3      IF .SELECT_DATA[0] NEQ 0
: 955      1085      3      OR .FOLDER_DESC[DSC$W_LENGTH] NEQ 0
: 956      1086      3      OR .START_FLAG OR .DIRFULL_FLAG
: 957      1087      3      THEN RETURN SIGNAL(MAIL$_NOTISAM);
: 958      1088      3      IF MAIL$FIND_MSG(1,.NEW_FILE)
: 959      1089      3      THEN
: 960      1090      3      MAIL$DIRECTORY(0);
: 961      1091      2      END;
: 962      1092      2
: 963      1093      2      DIR_MODE = 1;
: 964      1094      2      IF .MAIL$NMSG_STS
: 965      1095      2      THEN
: 966      1096      2      DIR_FLAG = 1;
: 967      1097      2      RETURN TRUE
: 968      1098      1      END;
```

! Display directory

! If more to display,

! then set flag

OFFC 00000

5B 00000000G 00 9E 00002
5A 00000000G 00 9E 00009.ENTRY MAIL\$CMD DIRECTORY, Save R2,R3,R4,R5,R6,R7,-; 0994
R8,R9,R10,R11
MOVAB MAIL\$NMSG_NUM, R11
MOVAB LIB\$SIGNAL, R10

59	00000000G	00	9E	00010	MOVAB	CL\$PRESENT, R9	:
58	00000000G	00	9E	00017	MOVAB	CL\$GET_VALUE, R8	:
57	00000000G	00	9E	0001E	MOVAB	MAIL\$NMSG, R7	:
56	00000000G	00	9E	00025	MOVAB	MAIL\$GL_FLAGS, R6	:
55	F8D6	CF	9E	0002C	MOVAB	MAIL\$OPEN_IF, R5	:
54	00000000	00	9E	00031	MOVAB	SELECT_FLAGS, R4	:
5E		0C	C2	00038	SUBL2	#12, SP	:
	FC	A4	B4	0003B	CLRW	DIR_FLAG	1019
	04	AE	7C	0003E	CLRW	FOLDER_DESC	1021
	34	A4	9F	00041	PUSHAB	DYN_DESC_3	1022
	F6CA	CF	9F	00044	PUSHAB	SD_FOLDER_NAME	:
68		02	FB	00048	CALLS	#2, CL\$GET_VALUE	:
15		50	E9	0004B	BLBC	R0, 1\$:
04	AE	34	A4	B0	MOVW	DYN_DESC_3, FOLDER_DESC	1024
08	AE	38	A4	D0	MOVL	DYN_DESC-3+4, FOLDER_DESC+4	1025
		04	AE	9F	PUSHAB	FOLDER_DESC	1027
0129	C5	01	FB	0005B	CALLS	#1, CHECK_FOLDER	:
	40	50	E9	00060	BLBC	STATUS, 3\$:
1D	A5	00	FB	00063	CALLS	#0, INIT_SELECT_DATA	1029
38	A5	00	FB	00067	CALLS	#0, PARSE_SELECTION_QUALS	1031
	35	50	E9	0006B	BLBC	STATUS, 3\$:
OC	64	04	E1	0006E	BBC	#4, SELECT_FLAGS, 2\$	1033
07	01	04	E0	00072	BBS	#4, MAIL\$GL_FLAGS+1, 2\$	1034
	00000000G	00	FB	00077	CALLS	#0, MAIL\$CLOSE_FILE	1035
		2C	A4	9F	PUSHAB	DYN_DESC_2	1037
	F821	CF	9F	00081	PUSHAB	SD_START	:
68		02	FB	00085	CALLS	#2, CL\$GET_VALUE	:
53		50	D0	00088	MOVL	R0, START_FLAG	:
	F6AB	CF	9F	0008B	PUSHAB	SD_FULL	1038
69		01	FB	0008F	CALLS	#1, CL\$PRESENT	:
FD	A4	50	90	00092	MOVB	R0, DIRFULL_FLAG	:
	F688	CF	9F	00096	PUSHAB	SD_FOLDER	1039
69		01	FB	0009A	CALLS	#1, CL\$PRESENT	:
17		50	E9	0009D	BLBC	R0, 5\$:
65		00	FB	000A0	CALLS	#0, MAIL\$OPEN_IF	1042
67		50	E9	000A3	BLBC	STATUS, 11\$:
03		66	E8	000A6	BLBS	MAIL\$GL_FLAGS, 4\$	1043
		31	000A9	23\$	BRW		:
	2C	A4	9F	000AC	PUSHAB	DYN_DESC_2	1045
00000000G	00	01	FB	000AF	CALLS	#1, MAIL\$DIR_FOLDER	:
	2B	04	000B6	RET			:
		53	E9	000B7	BLBC	START_FLAG, 7\$	1048
		5E	DD	000BA	PUSHL	SP	1052
	30	A4	DD	000BC	PUSHL	DYN_DESC_2+4	:
7E	2C	A4	3C	000BF	MOVZWL	DYN_DESC-2, -(SP)	:
00000000G	00	03	FB	000C3	CALLS	#3, LIB\$CVT_DTB	:
0F		50	E8	000CA	BLBS	STATUS, 6\$:
	2C	A4	9F	000CD	PUSHAB	DYN_DESC_2	:
		01	DD	000D0	PUSHL	#1	:
6A	007E1112	8F	DD	000D2	PUSHL	#8261906	:
		03	FB	000D8	CALLS	#3, LIB\$SIGNAL	:
		04	000DB	RET			:
		6E	D5	000DC	TSTL	START_MSGNUM	1053
		07	12	000DE	BNEQ	8\$:
6E		01	D0	000E0	MOVL	#1, START_MSGNUM	1054
		02	11	000E3	BRB	8\$	1048
		6E	D4	000E5	CLRL	START_MSGNUM	1056

52	01	A6	01	04	50	D4	000E7	8\$:	CLRL	R0	1059
			52		AE	B5	000E9		TSTW	FOLDER_DESC	
					02	13	000EC		BEQL	9\$	
					50	D6	000EE		INCL	R0	
					03	EF	000F0	9\$:	EXTZV	#3, #1, MAIL\$GL_FLAGS+1, R2	
					50	CA	000F6		BICL2	R0, R2	
					50	D4	000F9		CLRL	R0	1060
					64	D5	000FB		TSTL	SELECT_DATA	
					02	13	000FD		BEQL	10\$	
			52		50	D6	000FF		INCL	R0	
			52		50	CA	00101	10\$:	BICL2	R0, R2	
			52		52	D2	00104		MCOML	R2, NEW FILE	
			06		52	E9	00107		BLBC	NEW_FILE, 12\$	
			65		00	FB	0010A		CALLS	#0, MAIL\$OPEN_IF	1062
			16		50	E9	0010D	11\$:	BLBC	STATUS, 14\$	
			7B		66	E9	00110	12\$:	BLBC	MAIL\$GL_FLAGS, 22\$	1063
			04		52	E8	00113		BLBS	NEW_FILE, 13\$	1065
					67	D5	00116		TSTL	MAIL\$SL_NMSG	1066
					10	12	00118		BNEQ	15\$	
					54	DD	0011A	13\$:	PUSHL	R4	1068
				08	AE	9F	0011C		PUSHAB	FOLDER_DESC	
		00000000G	00		02	FB	0011F		CALLS	#2, MAIL\$SELECT_MSGS	
			01		50	E8	00126	14\$:	BLBS	STATUS, 15\$	
						04	00129		RET		
					67	D5	0012A	15\$:	TSTL	MAIL\$SL_NMSG	1069
					08	12	0012C		BNEQ	16\$	
				007E804A	8F	DD	0012E		PUSHL	#8290378	1070
					6E	11	00134		BRB	24\$	
					6E	D5	00136	16\$:	TSTL	START_MSGNUM	1071
					1D	12	00138		BNEQ	17\$	
			34		52	E8	0013A		BLBS	NEW FILE, 20\$	1072
			01		6B	D1	0013D		CMPL	MAIL\$SL_MSG_NUM, #1	1073
					2F	15	00140		BLEQ	20\$	
51			67	00000000G	00	C3	00142		SUBL3	MAIL\$SL_OUTLINES, MAIL\$SL_NMSG, R1	1076
			51		08	C0	0014A		ADDL2	#8, R1	
			50		6B	D0	0014D		MOVL	MAIL\$SL_MSG_NUM, R0	
			51		50	D1	00150		CMPL	R0, R1	
					15	14	00153		BGTR	18\$	
					16	11	00155		BRB	19\$	1075
51			67	00000000G	00	C3	00157	17\$:	SUBL3	MAIL\$SL_OUTLINES, MAIL\$SL_NMSG, R1	1078
			51		08	C0	0015F		ADDL2	#8, R1	
			50		6E	D0	00162		MOVL	START_MSGNUM, R0	
			51		50	D1	00165		CMPL	R0, RT	
					03	15	00168		BLEQ	19\$	
			50		51	D0	0016A	18\$:	MOVL	R1, R0	
					50	D5	0016D	19\$:	TSTL	R0	1077
					03	14	0016F		BGTR	21\$	
			50		01	D0	00171	20\$:	MOVL	#1, R0	
			6E		50	D0	00174	21\$:	MOVL	R0, START_MSGNUM	
					01	DD	00177		PUSHL	#1	1080
					01	DD	00179		PUSHL	#1	
				08	AE	DD	0017B		PUSHL	START MSGNUM	
		00000000G	00		03	FB	0017E		CALLS	#3, MAIL\$FIND_I_MSG	
			37		50	E9	00185		BLBC	R0, 27\$	
			7E		FD	A4	9A	00188	MOVZBL	DIRFULL_FLAG, -(SP)	1081
					2A	11	0018C		BRB	26\$	
					64	D5	0018E	22\$:	TSTL	SELECT_DATA	1084

MAIL\$CMDS
V04-000

VAX/VMS PERSONAL MAIL UTILITY COMMANDS

E 8
16-Sep-1984 00:47:25
14-Sep-1984 12:42:16

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[MAIL.SRC]CMDS.B32;1

Page 42
(16)

		04	0C	12	00190	BNEQ	23\$:	
			AE	B5	00192	TSTW	FOLDER_DESC	:	1085
			07	12	00195	BNEQ	23\$:	
	04		53	E8	00197	BLBS	START_FLAG, 23\$:	1086
	0A	FD	A4	E9	0019A	BLBC	DIRFUL_FLAG, 25\$:	
	6A	007E80AA	8F	DD	0019E	PUSHL	#8290474	:	1087
			01	FB	001A4	CALLS	#1, LIB\$SIGNAL	:	
				04	001A7	RET		:	
			52	DD	001A8	PUSHL	NEW_FILE	:	1088
			01	DD	001AA	PUSHL	#1	:	
00000000G	00		02	FB	001AC	CALLS	#2, MAIL\$FIND_MSG	:	
	09		50	E9	001B3	BLBC	R0, 27\$:	
			7E	D4	001B6	CLRL	-(SP)	:	1090
00000000G	00		01	FB	001B8	CALLS	#1, MAIL\$DIRECTORY	:	
FE	A4		01	90	001BF	MOVB	#1, DIR_MODE	:	1093
	04	00000000G	00	E9	001C3	BLBC	MAIL\$MSG_STS, 28\$:	1094
FC	A4		01	90	001CA	MOVB	#1, DIR_FLAG	:	1096
	50		01	D0	001CE	MOVL	#1, R0	:	1097
			04	001D1	RET			:	1098

; Routine Size: 466 bytes, Routine Base: \$CODE\$ + 097E

; 969 1099 1


```
: 971      1100 1 GLOBAL ROUTINE MAIL$CMD_SELECT =
: 972      1101 1 ++
: 973      1102 1 FUNCTIONAL DESCRIPTION:
: 974      1103 1
: 975      1104 1     Select new messages, tell how many were selected.
: 976      1105 1
: 977      1106 1 --
: 978      1107 2 BEGIN
: 979      1108 2 LOCAL
: 980      1109 2     FOLDER_DESC : $BBLOCK[DSC$C_S_BLN],
: 981      1110 2     NEW_FILE;
: 982      1111 2
: 983      1112 2 INIT_SDESC(FOLDER_DESC);
: 984      1113 2 IF C[ISGET_VALUE(SD_FOLDER_NAME,DYN_DESC_3)
: 985      1114 3 THEN BEGIN
: 986      1115 3     FOLDER_DESC[DSC$W_LENGTH] = .DYN_DESC_3[DSC$W_LENGTH];
: 987      1116 3     FOLDER_DESC[DSC$A_POINTER] = .DYN_DESC_3[DSC$A_POINTER];
: 988      P 1117 3     IF_ERR(CHECK_FOLDER(FOLDER_DESC));
: 989      1118 3     RETURN .STATUS;
: 990      1119 2 END;
: 991      1120 2 INIT_SELECT_DATA();
: 992      P 1121 2 IF_ERR(PARSE_SELECTION_QUALS());
: 993      1122 2     RETURN .STATUS;
: 994      1123 2
: 995      1124 2 IF .SELECT_FLAGS[QUAL_NEW]
: 996      1125 2     AND NOT .MAIL$GL_FLAGS[MAIF_V_DFMSG]
: 997      1126 2     THEN MAIL$CLOSE_FILE();
: 998      1127 2
: 999      1128 3 IF (NEW_FILE = NOT .MAIL$GL_FLAGS[MAIF_V_FILOPN]
1000      1129 4     OR (.FOLDER_DESC[DSC$W_LENGTH] NEQ 0)
1001      1130 3     OR (.SELECT_DATA[0] NEQ 0))
1002      1131 2     OR .MAIL$L_NMSG EQL 0
1003      P 1132 2 THEN IF_ERR(MAIL$OPEN_IF());
1004      1133 2     RETURN .STATUS;
1005      1134 2 IF .MAIL$GL_FLAGS[MAIF_V_ISAM]
1006      1135 3 THEN BEGIN
1007      P 1136 3     IF_ERR(MAIL$SELECT_MSGS(FOLDER_DESC,SELECT_DATA));
1008      1137 3     RETURN .STATUS;
1009      1138 3     SIGNAL(MAIL$_SELECTED,1,.MAIL$L_NMSG);
1010      1139 3 END
1011      1140 2 ELSE SIGNAL(MAIL$_NOTISAM);
1012      1141 2
1013      1142 2 RETURN TRUE
: 1014      1143 1 END;
```

```
007C 00000
56 00000000G 00 9E 00002
55 00000000G 00 9E 00009
54 00000000G 00 9E 00010
53      F525 CF 9E 00017
52 00000000' 00 9E 0001C
      7E 7C 00023
      34 A2 9F 00025
```

```
.ENTRY MAIL$CMD_SELECT, Save R2,R3,R4,R5,R6
MOVAB LIB$SIGNAL, R6
MOVAB MAIL$L_NMSG, R5
MOVAB MAIL$GL_FLAGS, R4
MOVAB SD_FOLDER_NAME, R3
MOVAB SELECT_FLAGS, R2
CLRQ FOLDER_DESC
PUSHAB DYN_DESC_3
```

```
: 1100
:
:
:
:
:
: 1112
: 1113
```

00000000G	00	53	DD	00028	PUSHL	R3	:
	13	02	FB	0002A	CALLS	#2, CLISGET_VALUE	:
	6E	50	E9	00031	BLBC	R0, 1\$:
04	AE	34	A2	B0 00034	MOVW	DYN_DESC_3, FOLDER_DESC	1115
		38	A2	D0 00038	MOVL	DYN_DESC_3+4, FOLDER_DESC+4	1116
F819	CF		5E	DD 0003D	PUSHL	SP	1118
	79		01	FB 0003F	CALLS	#1, CHECK_FOLDER	:
F705	CF		50	E9 00044	BLBC	STATUS, 9\$:
F71B	CF		00	FB 00047 1\$:	CALLS	#0, INIT_SELECT_DATA	1120
	6C		00	FB 0004C	CALLS	#0, PARSE_SELECTION_QUALS	1122
0C	62		50	E9 00051	BLBC	STATUS, 9\$:
07	A4		04	E1 00054	BBC	#4, SELECT_FLAGS, 2\$	1124
	00		04	E0 00058	BBS	#4, MAIL\$GL_FLAGS+1, 2\$	1125
00000000G	00		00	FB 0005D	CALLS	#0, MAIL\$CLOSE_FILE	1126
			50	D4 00064 2\$:	CLRL	R0	1129
			6E	B5 00066	TSTW	FOLDER_DESC	:
			02	13 00068	BEQL	3\$:
51	01	A4	50	D6 0006A	INCL	R0	:
	50	01	03	EF 0006C 3\$:	EXTZV	#3, #1, MAIL\$GL_FLAGS+1, R1	:
		51	50	CB 00072	BICL3	R0, R1, R0	:
			51	D4 00076	CLRL	R1	1130
			62	D5 00078	TSTL	SELECT_DATA	:
			02	13 0007A	BEQL	4\$:
			51	D6 0007C	INCL	R1	:
	50		51	CA 0007E 4\$:	BICL2	R1, R0	:
	50		50	D2 00081	MCOML	R0, NEW_FILE	:
	04		50	E8 00084	BLBS	NEW_FILE, 5\$:
			65	D5 00087	TSTL	MAIL\$GL_NMSG	1131
			08	12 00089	BNEQ	6\$:
F6A4	CF		00	FB 0008B 5\$:	CALLS	#0, MAIL\$OPEN_IF	1133
	2D		50	E9 00090	BLBC	STATUS, 9\$:
	1E		64	E9 00093 6\$:	BLBC	MAIL\$GL_FLAGS, 7\$	1134
			52	DD 00096	PUSHL	R2	1137
		04	AE	9F 00098	PUSHAB	FOLDER_DESC	:
00000000G	00		02	FB 0009B	CALLS	#2, MAIL\$SELECT_MSGS	:
	1B		50	E9 000A2	BLBC	STATUS, 9\$:
			65	DD 000A5	PUSHL	MAIL\$GL_NMSG	1138
			01	DD 000A7	PUSHL	#1	:
	007EBE83		8F	DD 000A9	PUSHL	#8306307	:
	66		03	FB 000AF	CALLS	#3, LIB\$SIGNAL	:
			09	11 000B2	BRB	8\$	1134
	007E80AA		8F	DD 000B4 7\$:	PUSHL	#8290474	1140
	66		01	FB 000BA	CALLS	#1, LIB\$SIGNAL	:
	50		01	D0 000BD 8\$:	MOVL	#1, R0	1142
			04	000C0 9\$:	RET		1143

; Routine Size: 193 bytes, Routine Base: \$CODE\$ + 0B50


```
1016 1144 1 GLOBAL ROUTINE MAIL$CMD_MOVE =
1017 1145 1 ++
1018 1146 1 FUNCTIONAL DESCRIPTION:
1019 1147 1
1020 1148 1 This is the MOVE command action routine and is called to move
1021 1149 1 the current message from one isam folder to another in the same file.
1022 1150 1
1023 1151 1 INPUTS:
1024 1152 1
1025 1153 1 Current message context.
1026 1154 1
1027 1155 1 OUTPUTS:
1028 1156 1
1029 1157 1 Current message is filed in the specified folder
1030 1158 1 --
1031 1159 2 BEGIN
1032 1160 2
1033 1161 2 LOCAL
1034 1162 2 STATUS,
1035 1163 2 SAVE_MSGNUM,
1036 1164 2 CONFIRM_FLAG,
1037 1165 2 COPY_FLAGS : $BBLOCK[4];
1038 1166 2
1039 1167 2 COPY_FLAGS = 0;
1040 1168 2 IF NOT .MAIL$GL_FLAGS[MAIF V_FILOPN]
1041 1169 2 THEN RETURN SIGNAL(MAIL$_NOTREADIN);
1042 1170 2
1043 1171 2 IF NOT .MAIL$GL_FLAGS[MAIF V_ISAM]
1044 1172 2 THEN RETURN SIGNAL(MAIL$_NOTISAM);
1045 1173 2
1046 P 1174 2 IF_ERR(CLISGET_VALUE(SD_FOLDER_NAME,DYN_DESC);,
1047 1175 2 RETURN .STATUS);
1048 1176 2
1049 1177 2 CONFIRM_FLAG = CLISPRESENT(SD_CONFIRM) AND 1;
1050 1178 2
1051 P 1179 2 IF_ERR(CHECK_FOLDER(DYN_DESC);,
1052 1180 2 RETURN .STATUS);
1053 1181 2
1054 1182 2 COPY_FLAGS[MCR V_ALL] = CLISPRESENT(SD_ALL) AND 1;
1055 1183 2 IF NOT .COPY_FLAGS[MCR V_ALL]
1056 1184 2 AND .MAIL$CURMSG_EQL 0
1057 1185 2 THEN RETURN SIGNAL(MAIL$_NOTREADIN);
1058 1186 2
1059 1187 2 IF .COPY_FLAGS[MCR V_ALL]
1060 1188 2 AND .MAIL$NMSG_EQL 0
1061 1189 2 THEN RETURN SIGNAL(MAIL$_NOMSGS);
1062 1190 2 CLISGET_VALUE(SD_FILE,DYN_DESC_2);
1063 1191 2
1064 1192 2 IF .DYN_DESC_2[DSC$_LENGTH] EQL 0
1065 1193 2 THEN BEGIN
1066 1194 2
1067 1195 2 Move to a folder in the same file
1068 1196 2
1069 1197 2 IF NOT .COPY_FLAGS[MCR V_ALL]
1070 1198 2 THEN MAIL$FILE_I_MSG(DYN_DESC,.CONFIRM_FLAG)
1071 1199 2 ELSE BEGIN
1072 1200 2 SAVE_MSGNUM = .MAIL$MSG_NUM;
```

```
: 1073      1201      4      IF .MAIL$L_NMSG GTRU 0
: 1074      1202      4      THEN INCRU-I FROM 1 TO .MAIL$L_NMSG
: 1075      1203      5      DO BEGIN
: 1076      1204      5          MAIL$L_CURMSG = .MAIL$L_MSGTBL[.I-1];
: 1077      1205      5          IF .MAIL$L_CURMSG NEQ 0
: 1078      1206      6              THEN BEGIN
: 1079      1207      6                  MAIL$L_MSG_NUM = .I;
: 1080      1208      6                  STATUS = MAIL$FILE I MSG(DYN_DESC,.CONFIRM_FLAG);
: 1081      1209      6                  IF .STATUS EQL RMS$_RNF
: 1082      1210      6                      THEN EXITLOOP;
: 1083      1211      5                  END;
: 1084      1212      4              END;
: 1085      1213      4          MAIL$L_MSG_NUM = .SAVE_MSGNUM;
: 1086      1214      4          IF .SAVE_MSGNUM NEQ 0
: 1087      1215      4              THEN MAIL$L_CURMSG = .MAIL$L_MSGTBL[.SAVE_MSGNUM-1];
: 1088      1216      3          END;
: 1089      1217      3      END
: 1090      1218      3      ELSE BEGIN
: 1091      1219      3          |
: 1092      1220      3          | Move to a folder in another file.
: 1093      1221      3          |
: 1094      1222      3          IF .CONFIRM_FLAG
: 1095      1223      3              THEN COPY_FLAGS[MCR V CONFNEW] = TRUE;
: 1096      1224      3          COPY_FLAGS[MCR V DELMSG] = TRUE;
: 1097      1225      3          MAIL$COPY_MSGSTDYN_DESC_2,DYN_DESC,.COPY_FLAGS,TELL_CREATED);
: 1098      1226      2          END;
: 1099      1227      2
: 1100      1228      2      RETURN TRUE
: 1101      1229      1      END;
```

			OFFC 00000		.ENTRY	MAILSCMD MOVE, Save R2,R3,R4,R5,R6,R7,R8,-	
						R9,R10,RT1	1144
					MOVAB	CLISGET_VALUE, R11	
					MOVAB	MAIL\$L_MSG_NUM, R10	
					MOVAB	MAIL\$L_CURMSG, R9	
					MOVAB	SD_FOLDER_NAME, R8	
					MOVAB	DYN_DESC, R7	
					CLRL	COPY_FLAGS	1167
	08 00000000G	00	55 D4 00023		BBS	#3, MAIL\$GL_FLAGS+1, 2\$	1168
		007E8022	8F DD 0002D 1\$:		PUSHL	#8290338	1169
			5D 11 00033		BRB	7\$	
		08 00000000G	00 E8 00035 2\$:		BLBS	MAIL\$GL_FLAGS, 3\$	1171
		007E80AA	8F DD 0003C		PUSHL	#8290474	1172
			4E 11 00042		BRB	7\$	
			57 DD 00044 3\$:		PUSHL	R7	1175
			58 DD 00046		PUSHL	R8	
	6B		02 FB 00048		CALLS	#2, CLISGET_VALUE	
	16		50 E9 0004B		BLBC	STATUS, 4\$	
		94	A8 9F 0004E		PUSHAB	SD_CONFIRM	1177
			01 FB 00051		CALLS	#1, CLISPRESENT	
56		50 00000000G	00 EF 00058		EXTZV	#0, #1, R0, CONFIRM_FLAG	
			01		PUSHL	R7	1180
		F738	57 DD 0005D				
			CF 01 FB 0005F		CALLS	#1, CHECK_FOLDER	

55	01	00000000G	00	FF74	50	E8	00064	4\$:	BLBS	STATUS, 5\$:		
	08		55		04	04	00067		RET		:		
	16		55		C8	9F	00068	5\$:	PUSHAB	SD_ALL	:	1182	
					01	FB	0006C		CALLS	#1, CLIS\$PRESENT	:		
					50	F0	00073		INSV	R0, #1, #1, COPY_FLAGS	:		
					01	E0	00078		BBS	#1, COPY_FLAGS, 8\$:	1183	
					69	D5	0007C		TSTL	MAIL\$L_CURMSG	:	1184	
					AD	13	0007E		BEQL	1\$:		
					01	E1	00080		BBC	#1, COPY_FLAGS, 8\$:	1187	
					00	D5	00084	6\$:	TSTL	MAIL\$L_NMSG	:	1188	
					0E	12	0008A		BNEQ	8\$:		
					8F	DD	0008C		PUSHL	#8290378	:	1189	
					01	FB	00092	7\$:	CALLS	#1, LIB\$SIGNAL	:		
						04	00099		RET		:		
					08	A7	9F	0009A	8\$:	PUSHAB	DYN_DESC_2	:	1190
					EC	A8	9F	0009D	PUSHAB	SD_FILE	:		
					6B	02	FB	000A0	CALLS	#2, CLIS\$GET_VALUE	:		
					08	A7	B5	000A3	TSTW	DYN_DESC_2	:	1192	
						61	12	000A6	BNEQ	14\$:		
					OD	01	E0	000A8	BBS	#1, COPY_FLAGS, 9\$:	1197	
						56	DD	000AC	PUSHL	CONFIRM_FLAG	:	1198	
						57	DD	000AE	PUSHL	R7	:		
						02	FB	000B0	CALLS	#2, MAIL\$FILE_I_MSG	:		
						6E	11	000B7	BRB	16\$:		
						6A	D0	000B9	9\$:	MOVL	MAIL\$L_MSG_NUM, SAVE_MSGNUM	:	1200
						00	D0	000BC	MOVL	MAIL\$L_NMSG, R4	:	1201	
						31	13	000C3	BEQL	13\$:		
						01	D0	000C5	MOVL	#1, I	:	1202	
						27	11	000C8	BRB	12\$:		
						00	D0	000CA	10\$:	MOVL	MAIL\$L_MSGTBL, R1	:	1204
						69	D0	000D1	MOVL	-4(R1)[I], MAIL\$L_CURMSG	:		
						17	13	000D6	BEQL	11\$:	1205	
						52	D0	000D8	MOVL	I, MAIL\$L_MSG_NUM	:	1207	
						56	DD	000DB	PUSHL	CONFIRM_FLAG	:	1208	
						57	DD	000DD	PUSHL	R7	:		
						02	FB	000DF	CALLS	#2, MAIL\$FILE_I_MSG	:		
						50	D1	000E6	CMPL	STATUS, #9899\$:	1209	
						07	13	000ED	BEQL	13\$:		
						52	D6	000EF	11\$:	INCL	I	:	1202
						52	D1	000F1	12\$:	CMPL	I, R4	:	
						D4	1B	000F4	BLEQU	10\$:		
						53	D0	000F6	13\$:	MOVL	SAVE_MSGNUM, MAIL\$L_MSG_NUM	:	1213
						2C	13	000F9	BEQL	16\$:	1214	
						00	D0	000FB	MOVL	MAIL\$L_MSGTBL, R0	:	1215	
						69	D0	00102	MOVL	-4(R0)[SAVE_MSGNUM], MAIL\$L_CURMSG	:		
						1E	11	00107	BRB	16\$:	1192	
						56	E9	00109	14\$:	BLBC	CONFIRM_FLAG, 15\$:	1222
						8F	A8	0010C	BISW2	#256, COPY_FLAGS	:	1223	
						8F	88	00111	15\$:	BISB2	#128, COPY_FLAGS	:	1224
						CF	9F	00115	PUSHAB	TELL_CREATED	:	1225	
						55	DD	00119	PUSHL	COPY_FLAGS	:		
						57	DD	0011B	PUSHL	R7	:		
						A7	9F	0011D	PUSHAB	DYN_DESC_2	:		
						04	FB	00120	CALLS	#4, MAIL\$COPY_MSGS	:		
						01	D0	00127	16\$:	MOVL	#1, R0	:	1228
						04	0012A		RET		:	1229	

MAIL\$CMDS
V04-000

VAX/VMS PERSONAL MAIL UTILITY COMMANDS

^{K 8}
16-Sep-1984 00:47:25
14-Sep-1984 12:42:16

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[MAIL.SRC]CMDS.B32;1 Page 48
(18)

; Routine Size: 299 bytes, Routine Base: \$CODE\$ + 0C11


```
: 1103      1230 1 GLOBAL ROUTINE MAIL$LOAD_LBRSHR =
: 1104      1231 1 ++
: 1105      1232 1 FUNCTIONAL DESCRIPTION:
: 1106      1233 1
: 1107      1234 1 Load LBRSHR if it has not been loaded
: 1108      1235 1
: 1109      1236 1 --
: 1110      1237 2 BEGIN
: 1111      1238 2
: 1112      1239 2 IF .MAIL$A_OUTPUT_HELP NEQ 0
: 1113      1240 2 THEN RETURN TRUE;
: 1114      1241 2
: 1115      1242 2 RETURN LIB$FIND_IMAGE_SYMBOL($DESCRIPTOR('LBRSHR'),
: 1116      1243 2 $DESCRIPTOR('LBR$OUTPUT_HELP'),
: 1117      1244 2 MAIL$A_OUTPUT_HELP)
: 1118      1245 1 END;
```

50	4C	45	48	5F	54	55	50	54	55	4F	24	52	42	4C	00D3C	P.ADH:	.ASCII	\LBRSHR\	:
															00D42		.BLKB	2	:
															00000006	P.ADG:	.LONG	6	:
															00000000		.ADDRESS	P.ADH	:
															00D48	P.ADJ:	.ASCII	\LBR\$OUTPUT_HELP\	:
															00D4C		.BLKB	1	:
															00D5B	P.ADI:	.LONG	15	:
															0000000F		.ADDRESS	P.ADJ	:
															00000000				:
															0004		.ENTRY	MAIL\$LOAD_LBRSHR, Save R2	: 1230
															00 9E	MOVAB	MAIL\$A_OUTPUT_HELP, R2	:	1239
															62 D5	TSTL	MAIL\$A_OUTPUT_HELP	:	1240
															04 13	BEQL	1\$:	1242
															01 D0	MOVL	#1, R0	:	1243
															04 00	RET		:	1245
															52 DD	PUSHL	R2	:	
															AF 9F	PUSHAB	P.ADI	:	
															AF 9F	PUSHAB	P.ADG	:	
															03 FB	CALLS	#3, LIB\$FIND_IMAGE_SYMBOL	:	
															04 00	RET		:	
															0020			:	

; Routine Size: 33 bytes, Routine Base: \$CODE\$ + 0D64

```

1120      1246 1 ROUTINE HELP_HANDLER (SIGARG,MECHARG) =
1121      1247 1 |++
1122      1248 1 | FUNCTIONAL DESCRIPTION:
1123      1249 1 |
1124      1250 1 |         Turn fatal signals from LBR$OUTPUT_HELP into errors
1125      1251 1 |
1126      1252 1 |--
1127      1253 2 BEGIN
1128      1254 2 MAP
1129      1255 2     SIGARG : REF $BBLOCK;
1130      1256 2
1131      1257 2 BIND
1132      1258 2     SIGNAME = SIGARG[CHF$L_SIG_NAME] : $BBLOCK;
1133      1259 2
1134      1260 2 IF .SIGNAME[STSSV_SEVERITY] EQL STSSK_SEVERE
1135      1261 2     THEN SIGNAME[STSSV_SEVERITY] = STSSK_ERROR;
1136      1262 2
1137      1263 2 RETURN SS$_RESIGNAL
1138      1264 1 END;

```

[illegible]

; Routine Size: 25 bytes, Routine Base: \$CODE\$ + 0D85


```
1140 1265 1 GLOBAL ROUTINE MAIL$CMD_HELP =
1141 1266 1 !++
1142 1267 1 FUNCTIONAL DESCRIPTION:
1143 1268 1
1144 1269 1 This is the HELP command action routine and is called to display
1145 1270 1 help for specified commands and topics.
1146 1271 1
1147 1272 1 INPUTS:
1148 1273 1
1149 1274 1 OUTPUTS:
1150 1275 1
1151 1276 1 Help for specified topic is displayed on SYS$OUTPUT.
1152 1277 1 --
1153 1278 2 BEGIN
1154 1279 2
1155 1280 2 BUILTIN
1156 1281 2 FP;
1157 1282 2
1158 1283 2 CLISGET VALUE(SD_HELPKEYS,COMMAND_DESC); !Get REST_OF_LINE
1159 1284 2 MAIL$CMD_ERASE(); !Erase the screen
1160 P 1285 2 IF_ERR(MAIL$LOAD_LBRSHR();,
1161 1286 2 RETURN TRUE);
1162 1287 2
1163 1288 2 MAIL$DISABLE_CTRLC();
1164 1289 2 .FP = HELP_HANDLER;
1165 1290 2 (.MAIL$A_OUTPUT_HELP)(MAIL$PUT_OUTPUT_0,
1166 1291 2 COMMAND_DESC,$DESCRIPTOR('MAILHELP'),
1167 1292 2 %REF(HLP$M_PROMPT OR HLP$M_PROCESS OR
1168 1293 2 HLP$M_GROUP OR HLP$M_SYSTEM OR
1169 1294 2 HLP$M_HELP),
1170 1295 2 MAIL$GET_INPUT);
1171 1296 2
1172 1297 2 MAIL$ENABLE_CTRLC();
1173 1298 2 RETURN TRUE
1174 1299 1 END;
```

```
50 4C 45 48 4C 49 41 4D 00D9E P.ADL: .ASCII \MAILHELP\
00DA6 .BLKB 2
00000008 00DA8 P.ADK: .LONG 8
00000000 00DAC .ADDRESS P.ADL
```

```
000C 00000 .ENTRY MAIL$CMD_HELP, Save R2,R3
53 00000000' 00 9E 00002 MOVAB COMMAND_DESC, R3
52 F3FB CF 9E 00009 MOVAB SD_HELPKEYS, R2
5E 04 C2 0000E SUBL2 #4, SP
0C BB 00011 PUSHR #^M<R2,R3>
00 02 FB 00013 CALLS #2, CLISGET VALUE
00000000G 00 00 FB 0001A CALLS #0, MAIL$CMD_ERASE
00000000G 00 00 FB 00021 CALLS #0, MAIL$LOAD_LBRSHR
8F AF 00 FB 00021 BLBC STATUS, 1$
37 50 E9 00025 CALLS #0, MAIL$DISABLE_CTRLC
00000000G 00 00 FB 00028 MOVAB HELP_HANDLER, (FP)
6D A3 AF 9E 0002F MOVAB MAIL$A_OUTPUT_HELP, R0
50 00000000' 00 D0 00033
```

1265
1283
1284
1286
1288
1289
1290

MAIL\$CMDS
V04-000

VAX/VMS PERSONAL MAIL UTILITY COMMANDS

B 9
16-Sep-1984 00:47:25
14-Sep-1984 12:42:16

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[MAIL.SRC]CMDS.B32;1

Page 52
(21)

04	AE	00000000G	00	9F	0003A	PUSHAB	MAIL\$GET_INPUT	:	
			2F	D0	00040	MOVL	#47, 4(SP)	:	1293
		04	AE	9F	00044	PUSHAB	4(SP)	:	1292
		0BF0	C2	9F	00047	PUSHAB	P,ADK	:	1291
			53	DD	0004B	PUSHL	R3	:	1290
			7E	D4	0004D	CLRL	-(SP)	:	
		00000000G	00	9F	0004F	PUSHAB	MAIL\$PUT_OUTPUT	:	
	60		06	FB	00055	CALLS	#6, (R0)-	:	
00000000G	00		00	FB	00058	CALLS	#0, MAIL\$ENABLE_CTRL	:	1297
	50		01	D0	0005F	MOVL	#1, R0	:	1298
			04	00062	1\$:	RET		:	1299

; Routine Size: 99 bytes, Routine Base: \$CODE\$ + 0DB0


```
: 1176      1300 1 GLOBAL ROUTINE MAIL$CMD_PRINT =
: 1177      1301 1 ++
: 1178      1302 1 FUNCTIONAL DESCRIPTION:
: 1179      1303 1
: 1180      1304 1 This is the PRINT command action routine and is called to queue
: 1181      1305 1 the current message for printing.
: 1182      1306 1
: 1183      1307 1 INPUTS:
: 1184      1308 1
: 1185      1309 1 Current message context.
: 1186      1310 1
: 1187      1311 1 OUTPUTS:
: 1188      1312 1
: 1189      1313 1 Current message is added to specified queue for printing.
: 1190      1314 1 --
: 1191      1315 2 BEGIN
: 1192      1316 2
: 1193      1317 2 LOCAL
: 1194      1318 2 N_COPIES,
: 1195      1319 2 PRQ_FLAGS : $BBLOCK[4];
: 1196      1320 2
: 1197      1321 2 PRQ_FLAGS = 0;
: 1198      1322 2 PRQ_FLAGS[PRQ_V_NOTIFY] = CL$PRESENT(SD_NOTIFY);
: 1199      1323 3 IF (PRQ_FLAGS[PRQ_V_FORCE] = CL$PRESENT(SD_PRINT))
: 1200      1324 2 THEN RETURN MAIL$PRINT(0,0,.PRQ_FLAGS);
: 1201      1325 2
: 1202      1326 3 IF NOT (PRQ_FLAGS[PRQ_V_ALL] = CL$PRESENT(SD_ALL))
: 1203      1327 3 THEN BEGIN
: 1204      P 1328 3 IF_ERR((IF .MAIL$GL_FLAGS[MAIF_V_ISAM]
: 1205      P 1329 3 THEN MAIL$REWIND_I-MSG()
: 1206      P 1330 3 ELSE MAIL$REWIND-MSG());, ! Rewind current message
: 1207      1331 3 RETURN SIGNAL(.STATUS));
: 1208      1332 3 END
: 1209      1333 3 ELSE BEGIN
: 1210      1334 4 IF (IF .MAIL$GL_FLAGS[MAIF_V_ISAM]
: 1211      1335 4 THEN .MAIL$_NMSG EQL 0
: 1212      1336 4 ELSE .MAIL$_MSGBUFSIZ EQL 0)
: 1213      1337 3 THEN RETURN SIGNAL(MAIL$_NOMSGS);
: 1214      1338 2 END;
: 1215      1339 2
: 1216      1340 2 N_COPIES = 1;
: 1217      1341 2 IF CL$GET_VALUE(SD_COPIES,COMMAND_DESC)
: 1218      P 1342 2 THEN IF_ERR(LIB$CVT_DTB(.COMMAND_DESC[DSC$W_LENGTH],
: 1219      P 1343 2 .COMMAND_DESC[DSC$A_POINTER],N_COPIES));,
: 1220      1344 2 RETURN SIGNAL(MAIL$_BADVALUE,1,COMMAND_DESC));
: 1221      1345 2
: 1222      1346 2 IF .N_COPIES EQL 0
: 1223      1347 2 OR .N_COPIES GTR 255
: 1224      1348 2 THEN RETURN SIGNAL(MAIL$_BADVALUE,1,COMMAND_DESC);
: 1225      1349 2
: 1226      1350 2 CL$GET_VALUE(SD_QUEUE,COMMAND_DESC);
: 1227      1351 2 MAIL$PRINT(COMMAND_DESC,.N_COPIES,.PRQ_FLAGS); !Queue the message(s)
: 1228      1352 2
: 1229      1353 2 RETURN TRUE;
: 1230      1354 1 END;
```

			07FC 00000	.ENTRY	MAIL\$CMD_PRINT, Save R2,R3,R4,R5,R6,R7,R8,-	
		5A	00000000G 00 9E 00002	MOVAB	R9,R10	1300
		59	00000000G 00 9E 00009	MOVAB	CLISGET VALUE, R10	
		58	00000000G 00 9E 00010	MOVAB	LIB\$SIGNAL, R9	
		57	00000000G 00 9E 00017	MOVAB	MAIL\$GL_FLAGS, R8	
		56	00000000G 00 9E 0001E	MOVAB	MAIL\$PRINT, R7	
		55	F2B0 CF 9E 00025	MOVAB	CLISPRESENT, R6	
		54	00000000' 00 9E 0002A	MOVAB	SD_NOTIFY, R5	
		5E	04 C2 00031	MOVAB	COMMAND_DESC, R4	
			53 D4 00034	SUBL2	#4, SP	
			55 DD 00036	CLRL	PRQ_FLAGS	1321
		66	01 FB 00038	PUSHL	R5	1322
53	01	00	50 FO 0003B	CALLS	#1, CLISPRESENT	
			A5 9F 00040	INSV	R0, #0, #1, PRQ_FLAGS	
		66	01 FB 00043	PUSHAB	SD_PRINT	1323
53	01	01	50 FO 00046	CALLS	#1, CLISPRESENT	
		08	50 E9 0004B	INSV	R0, #1, #1, PRQ_FLAGS	
			53 DD 0004E	BLBC	R0, 1\$	
			7E 7C 00050	PUSHL	PRQ_FLAGS	1324
		67	03 FB 00052	CLRL	-(SP)	
			04 00055	CALLS	#3, MAIL\$PRINT	
			C5 9F 00056 1\$:	RET		
		66	01 FB 0005A	PUSHAB	SD_ALL	1326
53	01	02	50 FO 0005D	CALLS	#1, CLISPRESENT	
		1A	50 E8 00062	INSV	R0, #2, #1, PRQ_FLAGS	
		09	68 E9 00065	BLBS	R0, 4\$	
		00	00 FB 00068	BLBC	MAIL\$GL_FLAGS, 2\$	1331
			07 11 0006F	CALLS	#0, MAIL\$REWIND_I_MSG	
		00	00 FB 00071 2\$:	BRB	3\$	
		21	50 E8 00078 3\$:	CALLS	#0, MAIL\$REWIND_MSG	
			50 DD 0007B	BLBS	STATUS, 8\$	
			19 11 0007D	PUSHL	STATUS	
		08	68 E9 0007F 4\$:	BRB	7\$	
			00 D5 00082	BLBC	MAIL\$GL_FLAGS, 5\$	1334
			06 11 0C088	TSTL	MAIL\$L_NMSG	1335
			00 D5 0J08A 5\$:	BRB	6\$	
			0A 12 00090 6\$:	TSTL	MAIL\$L_MSGBUFSIZ	1336
			8F DD 00092	BNEQ	8\$	
		69	01 FB 00098 7\$:	PUSHL	#8290378	1337
			04 0009B	CALLS	#1, LIB\$SIGNAL	
		6E	01 D0 0009C 8\$:	RET		
			54 DD 0009F	MOVL	#1, N_COPIES	1340
			C5 9F 000A1	PUSHL	R4	1341
		6A	02 FB 000A5	PUSHAB	SD_COPIES	
		12	50 E9 000A8	CALLS	#2, CLISGET_VALUE	
			5E DD 000AB	BLBC	R0, 9\$	
			A4 DD 000AD	PUSHL	SP	1344
		7E	64 3C 000B0	PUSHL	COMMAND_DESC+4	
		00	03 FB 000B3	MOVZWL	COMMAND_DESC, -(SP)	
		0E	50 E9 000BA	CALLS	#3, LIB\$CVT_DTB	
		52	6E D0 000BD 9\$:	BLBC	STATUS, 10\$	
			09 13 000C0	MOVL	N_COPIES, R2	1346
		000000FF	52 D1 000C2	BEQL	10\$	
				CMPL	R2, #255	1347

MAIL\$CMDS
V04-000

VAX/VMS PERSONAL MAIL UTILITY COMMANDS

E 9
16-Sep-1984 00:47:25
14-Sep-1984 12:42:16

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[MAIL.SRC]CMDS.B32;1

Page 55
(22)

		0E	1B	000C9		BLEQU	11\$:	
		54	DD	000CB	10\$:	PUSHL	R4	:	1348
		01	DD	000CD		PUSHL	#1	:	
	007E1112	8F	DD	000CF		PUSHL	#8261906	:	
69		03	FB	000D5		CALLS	#3, LIB\$SIGNAL	:	
			04	000D8		RET		:	
		54	DD	000D9	11\$:	PUSHL	R4	:	1350
	5C	A5	9F	000DB		PUSHAB	SD_QUEUE	:	
6A		02	FB	000DE		CALLS	#2, CLISGET_VALUE	:	
		0C	BB	000E1		PUSHR	#^M<R2,R3>	:	1351
		54	DD	000E3		PUSHL	R4	:	
67		03	FB	000E5		CALLS	#3, MAIL\$PRINT	:	
50		01	D0	000E8		MOVL	#1, R0	:	1353
			04	000EB		RET		:	1354

; Routine Size: 236 bytes, Routine Base: \$CODE\$ + 0E13

```
1232 1355 1 GLOBAL ROUTINE MAIL$CMD_NEXT =
1233 1356 1 ++
1234 1357 1 FUNCTIONAL DESCRIPTION:
1235 1358 1
1236 1359 1 This is the NEXT command action routine and is called to display
1237 1360 1 the text of the next message.
1238 1361 1
1239 1362 1 INPUTS:
1240 1363 1
1241 1364 1 Current message context.
1242 1365 1
1243 1366 1 OUTPUTS:
1244 1367 1
1245 1368 1 Next message text is displayed on SYS$OUTPUT.
1246 1369 1 --
1247 1370 2 BEGIN
1248 1371 2 LOCAL
1249 1372 2 EDIT_FLAG;
1250 1373 2
1251 1374 2 MAIL$CLR_DIR_FLAG();
1252 1375 2 EDIT_FLAG = CLISP$PRESENT(SD_EDIT);
1253 1376 2 IF .MAIL$GL_FLAGS[MAIF_V_ISAM]
1254 1377 3 THEN BEGIN
1255 1378 3 IF MAIL$FIND_I_MSG(0)
1256 1379 4 THEN (MAIL$REWIND_I_MSG(); READ_MESSAGE(.EDIT_FLAG))
1257 1380 3 ELSE IF .MAIL$L_MSG_STS EQL MAIL$NOMOREMSG
1258 1381 3 OR .MAIL$L_MSG_STS EQL MAIL$NOMSGS
1259 1382 3 THEN SIGNAL(.MAIL$L_MSG_STS);
1260 1383 3 END
1261 1384 2 ELSE
1262 1385 2 IF MAIL$FIND_MSG(0) ! find next message
1263 1386 2 THEN READ_MESSAGE(.EDIT_FLAG) ! Display it
1264 1387 2 ELSE
1265 1388 2 IF .MAIL$L_MSG_STS EQL MAIL$NOMOREMSG
1266 1389 2 THEN
1267 1390 2 SIGNAL(MAIL$NOMOREMSG);
1268 1391 2 RETURN TRUE
1269 1392 2
1270 1393 1 END;
```

```
001C 00000
54 00000000G 00 9E 00002
53 F4FF CF 9E 00009
63 00 FB 0000E
F294 CF 9F 00011
00000000G 00 01 FB 00015
52 50 D0 0001C
2E 00000000G 00 E9 0001F
7E D4 00026
00000000G 00 01 FB 00028
09 50 E9 0002F
00000000G 00 00 FB 00032
25 11 00039
```

```
.ENTRY MAIL$CMD_NEXT, Save R2,R3,R4
MOVAB MAIL$L_MSG_STS, R4
MOVAB MAIL$CLR_DIR_FLAG, R3
CALLS #0, MAIL$CLR_DIR_FLAG
PUSHAB SD_EDIT
CALLS #1, CLISP$PRESENT
MOVL R0, EDIT_FLAG
BLBC MAIL$GL_FLAGS, 3$
CLRL -(SP)
CALLS #1, MAIL$FIND_I_MSG
BLBC R0, 1$
CALLS #0, MAIL$REWIND_I_MSG
BRB 4$
```

```
: 1355
:
: 1374
: 1375
:
: 1376
: 1378
:
: 1379
:
```


MAIL\$CMDS
V04-000

VAX/VMS PERSONAL MAIL UTILITY COMMANDS

G 9
16-Sep-1984 00:47:25
14-Sep-1984 12:42:16

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[MAIL.SRC]CMDS.B32;1

Page 57
(23)

007E8042	50	64	D0	0003B	1\$:	MOVL	MAIL\$L_MSG_STS, R0	:	1380
	8F	50	D1	0003E		CMPL	R0, #8290370	:	
007E804A	8F	09	13	00045		BEQL	2\$:	
		50	D1	00047		CMPL	R0, #8290378	:	1381
		2E	12	0004E		BNEQ	7\$:	
		50	DD	00050	2\$:	PUSHL	R0	:	1382
		23	11	00052		BRB	6\$:	
00000000G	00	7E	D4	00054	3\$:	CLRL	-(SP)	:	1385
	08	01	FB	00056		CALLS	#1, MAIL\$FIND_MSG	:	
		50	E9	0005D		BLBC	R0, 5\$:	
85	A3	52	DD	00060	4\$:	PUSHL	EDIT_FLAG	:	1387
		01	FB	00062		CALLS	#1, READ_MESSAGE	:	
007E8042	8F	16	11	00066		BRB	7\$:	
		64	D1	00068	5\$:	CMPL	MAIL\$L_MSG_STS, #8290370	:	1389
		0D	12	0006F		BNEQ	7\$:	
	007E8042	8F	DD	00071		PUSHL	#8290370	:	1391
00000000G	00	01	FB	00077	6\$:	CALLS	#1, LIB\$SIGNAL	:	
	50	01	D0	0007E	7\$:	MOVL	#1, R0	:	1392
		04	00081			RET		:	1393

; Routine Size: 130 bytes, Routine Base: \$CODE\$ + 0EFF

```
: 1272      1394 1 GLOBAL ROUTINE MAIL$CMD_PURGE =
: 1273      1395 1 ++
: 1274      1396 1 FUNCTIONAL DESCRIPTION:
: 1275      1397 1
: 1276      1398 1 Delete all messages in the wastebasket folder, or if PURGE/RECLAIM
: 1277      1399 1 is specified, reclaim deleted space in the ISAM file
: 1278      1400 1
: 1279      1401 1 --
: 1280      1402 2 BEGIN
: 1281      1403 2
: 1282 P 1404 2 IF_ERR(MAIL$OPEN_IF());
: 1283      1405 2 RETURN .STATUS;
: 1284      1406 2
: 1285      1407 2 IF NOT .MAIL$GL_FLAGS[MAIF V ISAM]
: 1286      1408 2 THEN RETURN SIGNAL(MAIL$_NOTISAM);
: 1287      1409 2
: 1288      1410 2 IF CL$PRESENT(SD_RECLAIM)
: 1289      1411 3 THEN BEGIN
: 1290      1412 3 MAIL$GL_FLAGS[MAIF V RECL] = 1;
: 1291      1413 3 MAIL$EMPTY_WASTEBASKET();
: 1292      1414 3 MAIL$GL_FLAGS[MAIF V RECL] = 0;
: 1293      1415 3 MAIL$RECLAIM(CL$PRESENT(SD_STATISTICS));
: 1294      1416 3 END
: 1295      1417 2 ELSE SIGNAL(MAIL$_DELMSG,1,MAIL$EMPTY_WASTEBASKET());
: 1296      1418 2
: 1297      1419 2 RETURN TRUE
: 1298      1420 1 END;
```

56	00000000G	00	9E	00002	.ENTRY MAIL\$CMD_PURGE, Save R2,R3,R4,R5,R6	: 1394
55	00000000G	00	9E	00009	MOVAB MAIL\$EMPTY_WASTEBASKET, R6	:
54	00000000G	00	9E	00010	MOVAB CL\$PRESENT, R5	:
53	F2E8	CF	9E	00017	MOVAB LIB\$SIGNAL, R4	:
52	00000000G	00	9E	0001C	MOVAB MAIL\$OPEN_IF, R3	:
63		00	FB	00023	MOVAB MAIL\$GL_FLAGS, R2	:
47		50	E9	00026	CALLS #0, MAIL\$OPEN_IF	: 1405
0A		62	E8	00029	BLBC STATUS, 4\$:
64	007E80AA	8F	DD	0002C	BLBS MAIL\$GL_FLAGS, 1\$: 1407
		01	FB	00032	PUSHL #829047Z	: 1408
			04	00035	CALLS #1, LIB\$SIGNAL	:
	F19D	CF	9F	00036 1\$:	RET	:
65		01	FB	0003A	PUSHAB SD_RECLAIM	: 1410
1D		50	E9	0003D	CALLS #1, CL\$PRESENT	:
62	80	8F	88	00040	BLBC R0, 2\$:
66		00	FB	00044	BISB2 #128, MAIL\$GL_FLAGS	: 1412
62	80	8F	8A	00047	CALLS #0, MAIL\$EMPTY_WASTEBASKET	: 1413
	F19C	CF	9F	0004B	BICB2 #128, MAIL\$GL_FLAGS	: 1414
65		01	FB	0004F	PUSHAB SD_STATISTICS	: 1415
		50	DD	00052	CALLS #1, CL\$PRESENT	:
00000000G	00	01	FB	00054	PUSHL R0	:
		10	11	0005B	CALLS #1, MAIL\$RECLAIM	: 1410
66		00	FB	0005D 2\$:	BRB 3\$: 1417
		50	DD	00060	CALLS #0, MAIL\$EMPTY_WASTEBASKET	:
					PUSHL R0	:

MAIL\$CMDS
V04-000

VAX/VMS PERSONAL MAIL UTILITY COMMANDS

I 9
16-Sep-1984 00:47:25
14-Sep-1984 12:42:16

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[MAIL.SRC]CMDS.B32;1

Page 59
(24)

007EBE8B	01	DD	00062	PUSHL	#1	:
	8F	DD	00064	PUSHL	#8306315	:
64	03	FB	0006A	CALLS	#3, LIB\$SIGNAL	:
50	01	DO	0006D	MOVL	#1, R0	: 1419
	04	00070	4\$:	RET		: 1420

; Routine Size: 113 bytes, Routine Base: \$CODE\$ + 0F81

```
1300 1421 1 GLOBAL ROUTINE MAIL$CMD_COMPRESS =
1301 1422 1 |++
1302 1423 1 | FUNCTIONAL DESCRIPTION:
1303 1424 1 |
1304 1425 1 |     Process the COMPRESS command.
1305 1426 1 |
1306 1427 1 | --
1307 1428 2 BEGIN
1308 1429 2 LOCAL
1309 1430 2     FLAGS : BITVECTOR[32],
1310 1431 2     DESC : VECTOR[2, LONG],
1311 1432 2     STATUS;
1312 1433 2
1313 1434 2 DESC[0] = DESC[1] = 0;
1314 1435 2
1315 1436 2 | If a file was specified, use that. Else, if a file is open,
1316 1437 2 | use that. Else, open the default file.
1317 1438 2
1318 1439 2 IF CL$PRESENT(SD_FILE)
1319 1440 2 THEN BEGIN
1320 1441 2     MAIL$CLOSE_FILE();
1321 1442 2     CL$GET_VALUE(SD_FILE, DYN_DESC);
1322 1443 2     IF _ERR(MAIL$OPEN_FILE(DYN_DESC));,
1323 1444 2     RETURN .STATUS;
1324 1445 2 END
1325 1446 2 ELSE IF NOT .MAIL$GL_FLAGS[MAIF V FILOPN]
1326 1447 2 THEN IF _ERR(MAIL$OPEN_FILE(DESC));,
1327 1448 2     RETURN .STATUS;
1328 1449 2 IF NOT .MAIL$GL_FLAGS[MAIF V ISAM]
1329 1450 2 THEN RETURN SIGNAL(MAIL$NOTISAM);
1330 1451 2
1331 1452 2 CL$GET_VALUE(SD_OUTPUT, DYN_DESC_2);
1332 1453 2 !CL$GET_VALUE(SD_FDL, DYN_DESC_3);
1333 1454 2 FLAGS = 0;
1334 1455 2 MAIL$COMPRESS(.FLAGS, DYN_DESC_2, DESC);
1335 1456 2 RETURN TRUE
1336 1457 1 END;
```

		001C 00000	.ENTRY MAIL\$CMD COMPRESS, Save R2,R3,R4	1421
54 00000000G	00	9E 00002	MOVAB CL\$GET_VALUE, R4	
53 F07D	CF	9E 00009	MOVAB SD_FILE, R3	
52 00000000'	00	9E 0000E	MOVAB DYN_DESC, R2	
		7E 7C 00015	CLRQ DESC	1434
		53 DD 00017	PUSHL R3	1439
00000000G	00	01 FB 00019	CALLS #1, CL\$PRESENT	
	12	50 E9 00020	BLBC R0, 1\$	
00000000G	00	00 FB 00023	CALLS #0, MAIL\$CLOSE_FILE	1441
		52 DD 0002A	PUSHL R2	1442
		53 DD 0002C	PUSHL R3	
64	02	FB 0002E	CALLS #2, CL\$GET_VALUE	
	52	DD 00031	PUSHL R2	1444
	0A	11 00033	BRB 2\$	
0C 00000000G	00	03 E0 00035 1\$:	BBS #3, MAIL\$GL_FLAGS+1, 3\$	1446

MAIL\$CMDS
V04-000

VAX/VMS PERSONAL MAIL UTILITY COMMANDS

K 9
16-Sep-1984 00:47:25
14-Sep-1984 12:42:16

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[MAIL.SRC]CMDS.B32;1

Page 61
(25)

00000000G	00	5E	DD	0003D	PUSHL	SP	:	1448
	32	01	FB	0003F	CALLS	#1, MAIL\$OPEN_FILE	:	
	0E	50	E9	00046	BLBC	STATUS, 5\$:	
	00000000G	00	E8	00049	BLBS	MAIL\$GL_FLAGS, 4\$:	1449
	007E80AA	8F	DD	00050	PUSHL	#829047\$:	1450
00000000G	00	01	FB	00056	CALLS	#1, LIB\$SIGNAL	:	
		04		0005D	RET		:	
	08	A2	9F	0005E	PUSHAB	DYN_DESC_2	:	1452
	0080	C3	9F	00061	PUSHAB	SD_OUTPUT	:	
64		02	FB	00065	CALLS	#2, CLISGET_VALUE	:	
		50	D4	00068	CLRL	FLAGS	:	1454
		5E	DD	0006A	PUSHL	SP	:	1455
	08	A2	9F	0006C	PUSHAB	DYN_DESC_2	:	
		50	DD	0006F	PUSHL	FLAGS	:	
00000000G	00	03	FB	00071	CALLS	#3, MAIL\$COMPRESS	:	
	50	01	D0	00078	MOVL	#1, R0	:	1456
		04		0007B	RET		:	1457
				5\$:			:	

; Routine Size: 124 bytes, Routine Base: \$CODE\$ + 0FF2

```
1338 1458 1 GLOBAL ROUTINE MAIL$CMD_SEARCH =
1339 1459 1 ++
1340 1460 1 FUNCTIONAL DESCRIPTION:
1341 1461 1
1342 1462 1 This routine is the SEARCH command action routine and is called to
1343 1463 1 search for a message containing the specified string.
1344 1464 1
1345 1465 1 INPUTS:
1346 1466 1
1347 1467 1 OUTPUTS:
1348 1468 1
1349 1469 1 A message containing the search string is displayed.
1350 1470 1 --
1351 1471 2 BEGIN
1352 1472 2
1353 1473 2 LOCAL
1354 1474 2
1355 1475 2 TEST_REC: BLOCK[8,BYTE], ! Test record descriptor
1356 1476 2 TEST_REC_BUFF: BLOCK[MAIL$K_INBUFSZ]; ! Test record buffer
1357 1477 2
1358 1478 2 MAIL$CLR_DIR_FLAG();
1359 1479 2
1360 P 1480 2 IF_ERR(SELECT DEFAULT();, ! Open file, select msgs if not
1361 1481 2 RETURN .STATUS);
1362 1482 2
1363 1483 2 TEST_REC[DSC$A_POINTER] = TEST_REC_BUFF; ! Init descriptor
1364 1484 2
1365 1485 2 If a string is specified, use it and reset search to first message.
1366 1486 2
1367 1487 2 IF CL$GET_VALUE(SD_SEARCH1,COMMAND_DESC)
1368 1488 2 THEN BEGIN
1369 1489 2 MAIL$STRIP_QUOTES(COMMAND_DESC,COMMAND_DESC);
1370 1490 2 MAIL$UPCASE(COMMAND_DESC); !Uppcase in case quoted...DCL didn't
1371 1491 2 SEARCH_STRING[DSC$W_LENGTH] = MIN(.COMMAND_DESC[DSC$W_LENGTH],
1372 1492 2 %ALLOCATION(SEARCH_STRING_BUFF));
1373 1493 2 CH$MOVE(.SEARCH_STRING[DSC$W_LENGTH],.COMMAND_DESC[DSC$A_POINTER],
1374 1494 2 SEARCH_STRING_BUFF);
1375 P 1495 2 IF_ERR((IF .MAIL$GL_FLAGS[MAIF_V_ISAM]
1376 P 1496 2 THEN MAIL$FIND_I_MSG(1,1) ! Start with first message
1377 P 1497 2 ELSE MAIL$FIND_MSG(1));
1378 P 1498 2 IF .MAIL$GL_FLAGS[MAIF_V_ISAM]
1379 P 1499 2 THEN SIGNAL(.STATUS);
1380 1500 2 RETURN TRUE);
1381 1501 2 END
1382 1502 2 ELSE
1383 P 1503 2 IF_ERR((IF .MAIL$GL_FLAGS[MAIF_V_ISAM]
1384 P 1504 2 THEN MAIL$FIND_I_MSG(0) ! Else, start with next message
1385 P 1505 2 ELSE MAIL$FIND_MSG(0));
1386 P 1506 2 IF .MAIL$GL_FLAGS[MAIF_V_ISAM]
1387 P 1507 2 THEN SIGNAL(.STATUS);
1388 1508 2 RETURN TRUE);
1389 1509 2
1390 1510 2 Search through the messages starting with the current.
1391 1511 2
1392 1512 2 IF .MAIL$GL_FLAGS[MAIF_V_ISAM]
1393 1513 2 THEN BEGIN
1394 1514 2 LOCAL
```



```
: 1395      1515 3      RECDISC : $BBLOCK[DSC$C_S_BLN],
: 1396      1516 3      MOREFLAG;
: 1397      1517 3
: 1398      1518 4      DO BEGIN
: 1399      1519 4      MAIL$REWIND I MSG();
: 1400      1520 4      WHILE MAIL$GET_I_MSG(RECDISC,MOREFLAG)
: 1401      1521 5      DO BEGIN
: 1402      1522 5      TEST REC[DSC$W_LENGTH] = MIN(.RECDISC[DSC$W_LENGTH],MAIL$K_INBUFFSZ);
: 1403      1523 5      CH$MOVE(.TEST_REC[DSC$W_LENGTH],.RECDISC[DSC$A_POINTER],
: 1404      1524 5      TEST_REC_BUFF);
: 1405      1525 5      UPCASE(TEST_REC);
: 1406      1526 5      IF NOT CH$FAIL(CH$FIND_SUB(.TEST_REC[DSC$W_LENGTH],
: 1407      1527 5      .TEST_REC[DSC$A_POINTER],
: 1408      1528 5      .SEARCH_STRING[DSC$W_LENGTH],
: 1409      1529 5      .SEARCH_STRING[DSC$A_POINTER]))
: 1410      1530 6      THEN BEGIN
: 1411      1531 6      MAIL$REWIND I MSG();
: 1412      1532 6      READ MESSAGE(0);
: 1413      1533 6      RETURN TRUE;
: 1414      1534 5      END;
: 1415      1535 4      END;
: 1416      1536 4      IF .MAIL$GL_FLAGS[MAIF_V_CTRLCLF] THEN EXITLOOP;
: 1417      1537 4      END
: 1418      1538 3      UNTIL NOT MAIL$FIND_I_MSG(0);
: 1419      1539 3      END
: 1420      1540 3      ELSE BEGIN
: 1421      1541 3      DO
: 1422      1542 4      DO BEGIN
: 1423      1543 4      IF .MAIL$GL_FLAGS[MAIF_V_CTRLCLF] THEN EXITLOOP;
: 1424      1544 4      TEST_REC[DSC$W_LENGTH] = MIN(.MAIL$W_MSG_RSIZ,MAIL$K_INBUFFSZ);
: 1425      1545 4      CH$MOVE(.TEST_REC[DSC$W_LENGTH],.MAIL$SL_MSG_RADR,TEST_REC_BUFF);
: 1426      1546 4      UPCASE(TEST_REC);
: 1427      1547 4      IF NOT CH$FAIL(CH$FIND_SUB(.TEST_REC[DSC$W_LENGTH],
: 1428      1548 4      .TEST_REC[DSC$A_POINTER],
: 1429      1549 4      .SEARCH_STRING[DSC$W_LENGTH],
: 1430      1550 4      .SEARCH_STRING[DSC$A_POINTER]))
: 1431      1551 5      THEN BEGIN
: 1432      1552 5      MAIL$REWIND MSG();
: 1433      1553 5      MAIL$READ();
: 1434      1554 5      RETURN TRUE;
: 1435      1555 4      END;
: 1436      1556 4      END
: 1437      1557 3      UNTIL NOT MAIL$GET_MSG()
: 1438      1558 3      UNTIL NOT MAIL$FIND_MSG(0);
: 1439      1559 2      END;
: 1440      1560 2      SIGNAL(MAIL$_NOTFOUND,1,SEARCH_STRING);
: 1441      1561 2
: 1442      1562 2
: 1443      1563 2      RETURN TRUE;
: 1444      1564 1      END;
```

OFFC 00000

.ENTRY MAIL\$CMD SEARCH, Save R2,R3,R4,R5,R6,R7,R8,-; 1458
R9,R10,RT1

	5B	00000000G	00	9E	00002	MOVAB	MAIL\$FIND_MSG, R11		
	5A	00000000G	00	9E	00009	MOVAB	MAIL\$FIND_I_MSG, R10		
	59	00000000G	00	9E	00010	MOVAB	MAIL\$UPCASE, R9		
	58	F382	CF	9E	00017	MOVAB	MAIL\$CLR_DIR_FLAG, R8		
	57	00000000G	00	9E	0001C	MOVAB	MAIL\$GL_FLAGS, R7		
	56	00000000G	00	9E	00023	MOVAB	COMMAND_DESC, R6		
	5E	F7EC	CE	9E	0002A	MOVAB	-2068(SP), SP		
	68		00	FB	0002F	CALLS	#0, MAIL\$CLR_DIR_FLAG	1478	
C1	A8		00	FB	00032	CALLS	#0, SELECT_DEFAULT	1481	
	01		50	E8	00036	BLBS	STATUS, 1\$		
				04	00039	RET			
FC	AD	OC	AE	9E	0003A	1\$: MOVAB	TEST_REC_BUFF, TEST_REC+4	1483	
			56	DD	0003F	PUSHL	R6	1487	
		F141	CF	9F	00041	PUSHAB	SD_SEARCH1		
00000000G	00		02	FB	00045	CALLS	#2, CLISGET_VALUE		
	3D		50	E9	0004C	BLBC	R0, 4\$		
			56	DD	0004F	PUSHL	R6	1489	
			56	DD	00051	PUSHL	R6		
00000000G	00		02	FB	00053	CALLS	#2, MAIL\$STRIP_QUOTES	1490	
			56	DD	0005A	PUSHL	R6		
	69		01	FB	0005C	CALLS	#1, MAIL\$UPCASE	1491	
	50		66	3C	0005F	MOVZWL	COMMAND_DESC, R0		
0080	8F		50	B1	00062	CMPW	R0, #128		
			04	1B	00067	BLEQU	2\$		
	50	80	8F	9A	00069	MOVZBL	#128, R0		
	08		50	B0	0006D	2\$: MOVW	R0, SEARCH_STRING		
			50	A6	00071	MOVL	COMMAND_DESC+4, R0	1493	
FF40	C6		50	A6	00075	MOVC3	SEARCH_STRING, (R0), SEARCH_STRING_BUFF		
			60	08	0007C	BLBC	MAIL\$GL_FLAGS, 3\$	1500	
			09	67	E9	0007F	PUSHL	#1	
				01	DD	00081	PUSHL	#1	
	6A		02	FB	00083	CALLS	#2, MAIL\$FIND_I_MSG		
			13	11	00086	BRB	7\$		
			01	DD	00088	3\$: PUSHL	#1		
			0C	11	0008A	BRB	6\$		
	07		67	E9	0008C	4\$: BLBC	MAIL\$GL_FLAGS, 5\$	1508	
			7E	D4	0008F	CLRL	-(SP)		
	6A		01	FB	00091	CALLS	#1, MAIL\$FIND_I_MSG		
			05	11	00094	BRB	7\$		
			7E	D4	00096	5\$: CLRL	-(SP)		
	6B		01	FB	00098	6\$: CALLS	#1, MAIL\$FIND_MSG		
	0E		50	E8	0009B	7\$: BLBS	STATUS, 8\$		
	6B		67	E9	0009E	BLBC	MAIL\$GL_FLAGS, 13\$		
			50	DD	000A1	PUSHL	STATUS		
00000000G	00		01	FB	000A3	CALLS	#1, LIB\$SIGNAL		
			60	11	000AA	BRB	13\$		
	6D		67	E9	000AC	8\$: BLBC	MAIL\$GL_FLAGS, 15\$	1512	
00000000G	00		00	FB	000AF	9\$: CALLS	#0, MAIL\$REWIND_I_MSG	1519	
			5E	DD	000B6	10\$: PUSHL	SP	1520	
		08	AE	9F	000B8	PUSHAB	RECDESC		
00000000G	00		02	FB	000BB	CALLS	#2, MAIL\$GET_I_MSG		
	49		50	E9	000C2	BLBC	R0, 14\$		
	50	04	AE	3C	000C5	MOVZWL	RECDESC, R0	1522	
0200	8F		50	B1	000C9	CMPW	R0, #512		
			05	1B	000CE	BLEQU	11\$		
	50	0200	8F	3C	000D0	MOVZWL	#512, R0		
F8	AD		50	B0	000D5	11\$: MOVW	R0, TEST_REC		

OC	AE	08	BE	F8	AD	28	000D9	MOV C3	TEST_REC, @RECDESC+4, TEST_REC_BUFF	1523	
				F8	AD	9F	000E0	PUSHAB	TEST_REC	1525	
		69			01	FB	000E3	CALLS	#1, MAIL\$UPCASE		
		54		08	A6	3C	000E6	MOVZWL	SEARCH_STRING, R4	1528	
FC	BD	50		OC	A6	D0	000EA	MOVL	SEARCH_STRING+4, R0	1529	
		60			54	39	000EE	MATCHC	R4, (R0), TEST_REC, @TEST_REC+4		
					03	13	000F5	BEQL	12\$		
		53			54	D0	000F7	MOVL	R4, R3		
		53			54	C2	000FA	SUBL2	R4, R3		
					B7	13	000FD	BEQL	10\$		
		00000000G	00		00	FB	000FF	CALLS	#0, MAIL\$REWIND_I_MSG	1531	
					7E	D4	00106	CLRL	-(SP)	1532	
		85	A8		01	FB	00108	CALLS	#1, READ_MESSAGE		
					63	11	0010C	BRB	18\$	1533	
			73	01	A7	E8	0010E	BLBS	MAIL\$GL_FLAGS+1, 21\$	1536	
					7E	D4	00112	CLRL	-(SP)	1538	
		6A			01	FB	00114	CALLS	#1, MAIL\$FIND_I_MSG		
		6B			50	E9	00117	BLBC	R0, 21\$		
					93	11	0011A	BRB	9\$		
		5D		01	A7	E8	0011C	BLBS	MAIL\$GL_FLAGS+1, 20\$	1543	
		50	00000000G		00	3C	00120	MOVZWL	MAIL\$W_MSG_RSIZ, R0	1544	
		0200	8F		50	B1	00127	CMPW	R0, #5T2		
					05	1B	0012C	BLEQU	16\$		
			50	0200	8F	3C	0012E	MOVZWL	#512, R0		
		F8	AD		50	B0	00133	MOVW	R0, TEST_REC		
			50	00000000G	00	D0	00137	MOVL	MAIL\$MSG_RADR, R0	1545	
			60		F8	AD	28	0013E	MOV C3	TEST_REC, (R0), TEST_REC_BUFF	1546
					F8	AD	9F	00144	PUSHAB	TEST_REC	
		69			01	FB	00147	CALLS	#1, MAIL\$UPCASE		
		54		08	A6	3C	0014A	MOVZWL	SEARCH_STRING, R4	1549	
		50		OC	A6	D0	0014E	MOVL	SEARCH_STRING+4, R0	1550	
FC	BD	60			54	39	00152	MATCHC	R4, (R0), TEST_REC, @TEST_REC+4		
					03	13	00159	BEQL	17\$		
		53			54	D0	0015B	MOVL	R4, R3		
		53			54	C2	0015E	SUBL2	R4, R3		
					10	13	00161	BEQL	19\$		
		00000000G	00		00	FB	00163	CALLS	#0, MAIL\$REWIND_MSG	1552	
		00000000G	00		00	FB	0016A	CALLS	#0, MAIL\$READ	1553	
					24	11	00171	BRB	22\$	1554	
		00000000G	00		00	FB	00173	CALLS	#0, MAIL\$GET_MSG	1557	
			9F		50	E8	0017A	BLBS	R0, 15\$		
					7E	D4	0017D	CLRL	-(SP)	1558	
		6B			01	FB	0017F	CALLS	#1, MAIL\$FIND_MSG		
		97			50	E8	00182	BLBS	R0, 15\$		
				08	A6	9F	00185	PUSHAB	SEARCH_STRING	1561	
					01	DD	00188	PUSHL	#1		
				007E8072	8F	DD	0018A	PUSHL	#8290418		
		00000000G	00		03	FB	00190	CALLS	#3, LIB\$SIGNAL		
			50		01	D0	00197	MOVL	#1, R0	1563	
					04	0019A	RET			1564	

; Routine Size: 411 bytes, Routine Base: \$CODE\$ + 106E


```
: 1446      1565 1 GLOBAL ROUTINE MAIL$CMD_CURRENT =
: 1447      1566 1 ++
: 1448      1567 1 FUNCTIONAL DESCRIPTION:
: 1449      1568 1
: 1450      1569 1 Redisplay the current message
: 1451      1570 1
: 1452      1571 1 --
: 1453      1572 2 BEGIN
: 1454      1573 2 LOCAL
: 1455      1574 2 EDIT_FLAG;
: 1456      1575 2
: 1457      1576 2 MAIL$CLR_DIR_FLAG();
: 1458      1577 2 EDIT_FLAG = CLIS$PRESENT(SD_EDIT);
: 1459      1578 2 IF .MAIL$GL_FLAGS[MAIF_V_ISAM]
: 1460      1579 3 THEN BEGIN
: 1461      1580 3 IF NOT .MAIL$GL_FLAGS[MAIF_V_FILOPN]
: 1462      1581 3 OR .MAIL$MSG_NUM EQL 0
: 1463      1582 3 THEN RETURN SIGNAL(MAIL$NOTREADIN);
: 1464      1583 3 MAIL$FIND_I_MSG(.MAIL$MSG_NUM);
: 1465      1584 3 IF NOT .MAIL$MSG_STS
: 1466      1585 3 THEN RETURN SIGNAL(.MAIL$MSG_STS);
: 1467      1586 3 MAIL$REWIND_I_MSG();
: 1468      1587 3 END
: 1469      1588 3 ELSE BEGIN
: 1470      1589 3 IF NOT MAIL$REWIND_MSG()
: 1471      1590 3 THEN RETURN SIGNAL(MAIL$NOTREADIN);
: 1472      1591 2 END;
: 1473      1592 2 READ_MESSAGE(.EDIT_FLAG);
: 1474      1593 2
: 1475      1594 2 RETURN TRUE
: 1476      1595 1 END;
```

			001C 00000	.ENTRY MAIL\$CMD CURRENT, Save R2,R3,R4	: 1565
	54 00000000G	00	9E 00002	MOVAB MAIL\$MSG_NUM, R4	: 1576
	53 F1F5	CF	9E 00009	MOVAB MAIL\$CLR_DIR_FLAG, R3	: 1577
	63 EF8A	00	FB 0000E	CALLS #0, MAIL\$CLR_DIR_FLAG	: 1578
		CF	9F 00011	PUSHAB SD_EDIT	: 1580
	00000000G	00	FB 00015	CALLS #1, CLIS\$PRESENT	: 1581
		52	D0 0001C	MOVL R0, EDIT_FLAG	: 1583
	2C 00000000G	00	E9 0001F	BLBC MAIL\$GL_FLAGS, 2\$: 1584
2E 00000000G	00	03	E1 00026	BBC #3, MAIL\$GL_FLAGS+1, 3\$: 1585
		64	D5 0002E	TSTL MAIL\$MSG_NUM	: 1586
		2A	13 C0030	BEQL 3\$: 1578
	00000000G	00	DD 00032	PUSHL MAIL\$MSG_NUM	: 1579
		01	FB 00034	CALLS #1, MAIL\$FIND_I_MSG	: 1580
	50 00000000G	00	D0 0003B	MOVL MAIL\$MSG_STS, R0	: 1581
	04	50	E8 00042	BLBS R0, 1\$: 1582
		50	DD 00045	PUSHL R0	: 1583
		19	11 00047	BRB 4\$: 1584
	00000000G	00	FB 00049 1\$:	CALLS #0, MAIL\$REWIND_I_MSG	: 1585
		18	11 00050	BRB 5\$: 1586
	00000000G	00	FB 00052 2\$:	CALLS #0, MAIL\$REWIND_MSG	: 1587
	0E	50	E8 00059	BLBS R0, 5\$: 1588

MAIL\$CMDS
V04-000

VAX/VMS PERSONAL MAIL UTILITY COMMANDS

D 10
16-Sep-1984 00:47:25
14-Sep-1984 12:42:16

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[MAIL.SRC]CMDS.B32;1

Page 67
(27)

00000000G	00	007E8022	8F	DD	0005C	3\$:	PUSHL	#8290338	:	1590
			01	FB	00062	4\$:	CALLS	#1, LIB\$SIGNAL	:	
				04	00069		RET		:	
			52	DD	0006A	5\$:	PUSHL	EDIT FLAG	:	1592
85	A3		01	FB	0006C		CALLS	#1, READ_MESSAGE	:	
	50		01	DO	00070		MOVL	#1, R0	:	1594
			04	00073			RET		:	1595

; Routine Size: 116 bytes, Routine Base: \$CODE\$ + 1209

```
: 1478      1596 1 GLOBAL ROUTINE MAIL$CMD_FIRST =
: 1479      1597 1 |++
: 1480      1598 1 | FUNCTIONAL DESCRIPTION:
: 1481      1599 1 |
: 1482      1600 1 |         Retype the first message in the folder
: 1483      1601 1 |
: 1484      1602 1 | --
: 1485      1603 2 BEGIN
: 1486      1604 2 LOCAL
: 1487      1605 2     EDIT_FLAG;
: 1488      1606 2
: 1489      1607 2 MAIL$CLR_DIR_FLAG();
: 1490      1608 2 IF_ERR(SELECT DEFAULT();,
: 1491      1609 2     RETURN .STATUS);
: 1492      1610 2
: 1493      1611 2 EDIT_FLAG = CLISPRESNT(SD EDIT);
: 1494      1612 2 IF .MAIL$GL_FLAGS[MAIF_V_ISAM]
: 1495      1613 3 THEN BEGIN
: 1496      1614 3     MAIL$FIND_I_MSG(1);
: 1497      1615 3     IF NOT .MAIL$SL_MSG_STS
: 1498      1616 3     THEN RETURN SIGNAL(.MAIL$SL_MSG_STS);
: 1499      1617 3     MAIL$REWIND_I_MSG();
: 1500      1618 3     END
: 1501      1619 2 ELSE IF_ERR(MAIL$FIND_MSG(1);,
: 1502      1620 2     RETURN TRUE);
: 1503      1621 2 READ MESSAGE(.EDIT_FLAG);
: 1504      1622 2 RETURN TRUE
: 1505      1623 1 END;
```

			000C	00000	.ENTRY	MAIL\$CMD_FIRST, Save R2,R3	: 1596
	53	F188	CF	9E 00002	MOVAB	MAIL\$CLR_DIR_FLAG, R3	: 1607
	63		00	FB 00007	CALLS	#0, MAIL\$CLR_DIR_FLAG	: 1609
C1	A3		00	FB 0000A	CALLS	#0, SELECT_DEFAULT	: 1611
	50		50	E9 0000E	BLBC	STATUS, 5\$: 1612
		EF16	CF	9F 00011	PUSHAB	SD_EDIT	: 1614
00000000G	00		01	FB 00015	CALLS	#1, CLISPRESNT	: 1615
	52		50	D0 0001C	MOVL	R0, EDIT_FLAG	: 1616
	26	00000000G	00	E9 0001F	BLBC	MAIL\$GL_FLAGS, 2\$: 1617
			01	DD 00026	PUSHL	#1	: 1618
00000000G	00		01	FB 00028	CALLS	#1, MAIL\$FIND_I_MSG	: 1619
	50	00000000G	00	D0 0002F	MOVL	MAIL\$SL_MSG_STS, R0	: 1620
	0A		50	E8 00036	BLBS	R0, 1\$: 1621
			50	DD 00039	PUSHL	R0	: 1622
00000000G	00		01	FB 0003B	CALLS	#1, LIB\$SIGNAL	: 1623
				04 00042	RET		: 1624
00000000G	00		00	FB 00043 1\$:	CALLS	#0, MAIL\$REWIND_I_MSG	: 1625
			0C	11 0004A	BRB	3\$: 1626
			01	DD 0004C 2\$:	PUSHL	#1	: 1627
00000000G	00		01	FB 0004E	CALLS	#1, MAIL\$FIND_MSG	: 1628
	06		50	E9 00055	BLBC	STATUS, 4\$: 1629
			52	DD 00058 3\$:	PUSHL	EDIT_FLAG	: 1630
85	A3		01	FB 0005A	CALLS	#1, READ_MESSAGE	: 1631
	50		01	D0 0005E 4\$:	MOVL	#1, R0	: 1632

MAIL\$CMDS
V04-000

VAX/VMS PERSONAL MAIL UTILITY COMMANDS

F 10
16-Sep-1984 00:47:25
14-Sep-1984 12:42:16

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[MAIL.SRC]CMDS.B32;1 Page 69
(28)

04 00061 5\$: RET

; 1623

; Routine Size: 98 bytes, Routine Base: \$CODE\$ + 127D

```
1507 1624 1 GLOBAL ROUTINE MAIL$CMD_LAST =
1508 1625 1 ++
1509 1626 1 FUNCTIONAL DESCRIPTION:
1510 1627 1
1511 1628 1 Type the last message in the folder
1512 1629 1
1513 1630 1 --
1514 1631 2 BEGIN
1515 1632 2 LOCAL
1516 1633 2 EDIT_FLAG;
1517 1634 2
1518 1635 2 MAIL$CLR DIR FLAG();
1519 P 1636 2 IF_ERR(SELECT DEFAULT();,
1520 1637 2 RETURN .STATUS);
1521 1638 2
1522 1639 2 EDIT_FLAG = CLISP$PRESENT(SD_EDIT);
1523 1640 2 IF .MAIL$GL_FLAGS[MAIF_V_ISAM]
1524 1641 3 THEN BEGIN
1525 1642 3 IF .MAIL$L_NMSG EQL 0
1526 1643 3 THEN RETURN SIGNAL(MAIL$NOMSGS);
1527 1644 3 MAIL$FIND I_MSG(.MAIL$L_NMSG);
1528 1645 3 IF NOT .MAIL$L_MSG_STS
1529 1646 3 THEN RETURN SIGNAL(.MAIL$L_MSG_STS);
1530 1647 3 MAIL$REWIND_I_MSG();
1531 1648 3 END
1532 P 1649 2 ELSE IF_ERR(MAIL$FIND MSG('X'7FFFFFFF));,
1533 1650 2 RETURN TRUE);
1534 1651 2 READ MESSAGE(.EDIT_FLAG);
1535 1652 2 RETURN TRUE
1536 1653 1 END;
```

			001C 00000	.ENTRY MAIL\$CMD_LAST, Save R2,R3,R4	1624
	54	00000000G	00 9E 00002	MOVAB MAIL\$L_NMSG, R4	
	53	F11F	CF 9E 00009	MOVAB MAIL\$CLR DIR_FLAG, R3	
	63		00 FB 0000E	CALLS #0, MAIL\$CLR DIR_FLAG	1635
C1	A3		00 FB 00011	CALLS #0, SELECT_DEFAULT	1637
	60		50 E9 00015	BLBC STATUS, 7\$	
		EEAD	CF 9F 00018	PUSHAB SD_EDIT	1639
00000000G	00		01 FB 0001C	CALLS #1, CLISP\$PRESENT	
	52		50 D0 00023	MOVL R0, EDIT_FLAG	
	32	00000000G	00 E9 00026	BLBC MAIL\$GL_FLAGS, 4\$	1640
			64 D5 0002D	TSTL MAIL\$L_NMSG	1642
		007E804A	08 12 0002F	BNEQ 1\$	
			8F DD 00031	PUSHL #8290378	1643
			15 11 00037	BRB 2\$	
00000000G	00		64 DD 00039 1\$:	PUSHL MAIL\$L_NMSG	1644
	50	00000000G	01 FB 0003B	CALLS #1, MAIL\$FIND_I_MSG	
	0A		00 D0 00042	MOVL MAIL\$L_MSG_STS, R0	1645
			50 E8 00049	BLBS R0, 3\$	
00000000G	00		50 DD 0004C	PUSHL R0	1646
			01 FB 0004E 2\$:	CALLS #1, LIB\$SIGNAL	
			04 00055	RET	
00000000G	00		00 FB 00056 3\$:	CALLS #0, MAIL\$REWIND_I_MSG	1647

MAIL\$CMDS
V04-000

VAX/VMS PERSONAL MAIL UTILITY COMMANDS

H 10
16-Sep-1984 00:47:25
14-Sep-1984 12:42:16

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[MAIL.SRC]CMDS.B32;1

Page 71
(29)

00000000G	00	7FFFFFFF	10	11	0005D	BRB	5\$:	1640
	06		8F	DD	0005F	PUSHL	#2147483647	:	1650
			01	FB	00065	CALLS	#1, MAIL\$FIND_MSG	:	
			50	E9	0006C	BLBC	STATUS, 6\$:	
85	A3		52	DD	0006F	PUSHL	EDIT_FLAG	:	1651
	50		01	FB	00071	CALLS	#1, READ_MESSAGE	:	
			01	DD	00075	MOVL	#1, R0	:	1652
			04	00078	7\$:	RET		:	1653

; Routine Size: 121 bytes, Routine Base: \$CODE\$ + 12DF

```
: 1538      1654 1 GLOBAL ROUTINE MAIL$CMD_SPAWN =
: 1539      1655 1 ++
: 1540      1656 1 FUNCTIONAL DESCRIPTION:
: 1541      1657 1
: 1542      1658 1     Spawn a subprocess
: 1543      1659 1
: 1544      1660 1 --
: 1545      1661 2 BEGIN
: 1546      1662 2
: 1547      1663 2 OWN
: 1548      1664 2     PROC_DESC : DYN_DESC_DECL,
: 1549      1665 2     INP_DESC : DYN_DESC_DECL,
: 1550      1666 2     OUT_DESC : DYN_DESC_DECL;
: 1551      1667 2
: 1552      1668 2 LOCAL
: 1553      1669 2     SUBPROCID,
: 1554      1670 2     PURGE_VEC : VECTOR[2, LONG],
: 1555      1671 2     SPAWN_FLAGS;
: 1556      1672 2
: 1557      1673 2 IF .MAIL$GL_FLAGS[MAIF V CAPTIVE]
: 1558      1674 2 THEN RETURN SIGNAL(MAIL$_CAPTIVE);
: 1559      1675 2
: 1560      1676 2 SPAWN_FLAGS = 0;
: 1561      1677 2 CLISGET_VALUE(SD_PROCESS, PROC_DESC);
: 1562      1678 2 CLISGET_VALUE(SD_INPUT, INP_DESC);
: 1563      1679 2 CLISGET_VALUE(SD_OUTPUT, OUT_DESC);
: 1564      1680 2 CLISGET_VALUE(SD_P1, COMMAND_DESC);
: 1565      1681 2
: 1566      1682 2 IF NOT CLISPRESENT(SD_WAIT)
: 1567      1683 2 THEN SPAWN_FLAGS = .SPAWN_FLAGS OR 1;
: 1568      1684 2
: 1569      1685 2 IF NOT CLISPRESENT(SD_SYMBOLS)
: 1570      1686 2 THEN SPAWN_FLAGS = .SPAWN_FLAGS OR 2;
: 1571      1687 2
: 1572      1688 2 IF NOT CLISPRESENT(SD_LOGICAL_NAMES)
: 1573      1689 2 THEN SPAWN_FLAGS = .SPAWN_FLAGS OR 4;
: 1574      1690 2
: 1575      1691 2 MAIL$DISABLE CTRLC();
: 1576      1692 2 IF (.SPAWN_FLAGS AND 1) EQL 0          !ONLY IF SPAWN/WAIT
: 1577      1693 2 THEN BEGIN
: 1578      1694 3     PURGE_VEC[0] = 0;
: 1579      1695 3     PURGE_VEC[1] = %X'7FFFFFFF';
: 1580      1696 3     $PURGE$(INADR=PURGE_VEC);
: 1581      1697 3     END;
: 1582      1698 2 IF _ERR(LIB$SPAWN(COMMAND_DESC, INP_DESC, OUT_DESC,
: 1583      1699 2     SPAWN_FLAGS, PROC_DESC, SUBPROCID));,
: 1584      1700 2     (MAP STATUS : $BBLOCK;
: 1585      1701 2     STATUS[ST$V SEVERITY] = ST$K_ERROR;
: 1586      1702 2     MAIL$ENABLE CTRLC();
: 1587      1703 2     RETURN SIGNAL(.STATUS)));;
: 1588      1704 2
: 1589      1705 2 MAIL$ENABLE CTRLC();
: 1590      1706 2 IF (.SPAWN_FLAGS AND 1) EQL 0          !SPAWN /WAIT
: 1591      1707 2 THEN MAIL$CMD_SHOW_NEW();
: 1592      1708 2
: 1593      1709 2 RETURN TRUE
: 1594      1710 1 END;
```



```
.PSECT $OWN$,NOEXE,2

00# 000D0 PROC_DESC:
02 0E 000D2 .BYTE 0[2]
      000D4 .BYTE 14, 2
      000D8 INP_DESC:
02 0E 000DA .BYTE 0[2]
      000DC .BYTE 14, 2
      000E0 OUT_DESC:
02 0E 000E2 .BYTE 0[2]
      000E4 .BYTE 14, 2
      .BLKB 4

.EXTRN SYSS$PURGWS

.PSECT $CODE$,NOWRT,2

00FC 00000 .ENTRY MAIL$CMD SPAWN, Save R2,R3,R4,R5,R6,R7
57 00000000G 00 9E 00002 MOVAB MAIL$ENABLE_CTRL, R7
56 00000000G 00 9E 00009 MOVAB CLIS$PRESENT, R6
55 00000000G 00 9E 00010 MOVAB CLIS$GET_VALUE, R5
54 EDC5 CF 9E 00017 MOVAB SD_PROCESS, R4
53 00000000' 00 9E 0001C MOVAB PROC_DESC, R3
5E 10 C2 00023 SUBL2 #16, SP
09 00000000G 00 06 E1 00026 BBC #6, MAIL$GL_FLAGS+1, 1$
      007E812A 8F DD 0002E PUSHL #8290602
      0095 31 00034 BRW 6$
      04 AE D4 00037 1$: CLRL SPAWN_FLAGS
      53 DD 0003A PUSHL R3
      54 DD 0003C PUSHL R4
65 02 FB 0003E CALLS #2, CLIS$GET_VALUE
      08 A3 9F 00041 PUSHAB INP_DESC
      FF2C C4 9F 00044 PUSHAB SD_INPUT
65 02 FB 00048 CALLS #2, CLIS$GET_VALUE
      10 A3 9F 0004B PUSHAB OUT_DESC
      C4 A4 9F 0004E PUSHAB SD_OUTPUT
65 02 FB 00051 CALLS #2, CLIS$GET_VALUE
      F0 A3 9F 00054 PUSHAB COMMAND_DESC
      D0 A4 9F 00057 PUSHAB SD_P1
65 02 FB 0005A CALLS #2, CLIS$GET_VALUE
      0134 C4 9F 0005D PUSHAB SD_WAIT
66 01 FB 00061 CALLS #1, CLIS$PRESENT
04 04 50 E8 00064 BLBS R0, 2$
      01 88 00067 BISB2 #1, SPAWN_FLAGS
      44 A4 9F 0006B 2$: PUSHAB SD_SYMBOLS
66 01 FB 0006E CALLS #1, CLIS$PRESENT
04 04 50 E8 00071 BLBS R0, 3$
      02 88 00074 BISB2 #2, SPAWN_FLAGS
      A4 A4 9F 00078 3$: PUSHAB SD_LOGICAL_NAMES
66 01 FB 0007B CALLS #1, CLIS$PRESENT
04 04 50 E8 0007E BLBS R0, 4$
      04 88 00081 BISB2 #4, SPAWN_FLAGS
```

MAIL\$CMDS
V04-000

VAX/VMS PERSONAL MAIL UTILITY COMMANDS

K 10
16-Sep-1984 00:47:25
14-Sep-1984 12:42:16

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[MAIL.SRC]CMDS.B32;1

Page 74
(30)

52

03

00000000G	00		00	FB	00085	4\$:	CALLS	#0, MAIL\$DISABLE_CTRL C	:	1691
	15	04	AE	E8	0008C		BLBS	SPAWN_FLAGS, 5\$:	1692
		08	AE	D4	00090		CLRL	PURGE_VEC	:	1694
OC	AE	7FFFFFFF	8F	D0	00093		MOVL	#2147483647, PURGE_VEC+4	:	1695
		08	AE	9F	0009B		PUSHAB	PURGE_VEC	:	1696
00000000G	00		01	FB	0009E		CALLS	#1, SY\$SPURGWS	:	
		4008	8F	BB	000A5	5\$:	PUSHR	#^M<R3, SP>	:	1703
		0C	AE	9F	000A9		PUSHAB	SPAWN_FLAGS	:	
		10	A3	9F	000AC		PUSHAB	OUT_DESC	:	
		08	A3	9F	000AF		PUSHAB	INP_DESC	:	
		F0	A3	9F	000B2		PUSHAB	COMMAND_DESC	:	
00000000G	00		06	FB	000B5		CALLS	#6, LIB\$SPAWN	:	
	52		50	D0	000BC		MOVL	R0, STATUS	:	
	12		52	E8	000BF		BLBS	STATUS, 7\$:	
	00		02	F0	000C2		INSV	#2, #0, #3, STATUS	:	
	67		00	FB	000C7		CALLS	#0, MAIL\$ENABLE_CTRL C	:	
			52	DD	000CA		PUSHL	STATUS	:	
00000000G	00		01	FB	000CC	6\$:	CALLS	#1, LIB\$SIGNAL	:	
	67			04	000D3		RET		:	
	07	04	00	FB	000D4	7\$:	CALLS	#0, MAIL\$ENABLE_CTRL C	:	1705
			AE	E8	000D7		BLBS	SPAWN_FLAGS, 8\$:	1706
00000000G	00		00	FB	000DB		CALLS	#0, MAIL\$CMD_SHOW_NEW	:	1707
	50		01	D0	000E2	8\$:	MOVL	#1, R0	:	1709
			04	000E5			RET		:	1710

; Routine Size: 230 bytes, Routine Base: \$CODE\$ + 1358


```
: 1596      1711 1 GLOBAL ROUTINE MAIL$CMD_ATTACH =
: 1597      1712 1 |++
: 1598      1713 1 | FUNCTIONAL DESCRIPTION:
: 1599      1714 1 |
: 1600      1715 1 |         Attach to a subprocess
: 1601      1716 1 |
: 1602      1717 1 |--
: 1603      1718 2 BEGIN
: 1604      1719 2
: 1605      1720 2 LOCAL
: 1606      1721 2     STATUS,
: 1607      1722 2     SUBPROCPID : VOLATILE,
: 1608      1723 2     PURGE_VEC : VECTOR[2, LONG],
: 1609      1724 2     PARENT_FLAG,
: 1610      1725 2     ITMLST : $ITMLST_DECL(ITEMS=1);
: 1611      1726 2
: 1612      1727 2 CLISGET_VALUE(SD_P1, COMMAND_DESC);
: 1613      1728 2
: 1614      1729 2 PARENT_FLAG = CLISPRESENT(SD_PARENT);
: 1615      1730 2
: 1616      1731 2 IF .PARENT_FLAG
: 1617      1732 2 THEN BEGIN
: 1618      P 1733 2     $ITMLST_INIT(ITMLST=ITMLST,
: 1619      1734 2     (ITMCOBJ=JPI$_OWNER, BUFADR=SUBPROCPID));
: 1620      1735 2
: 1621      1736 2     STATUS = $GETJPIW(ITMLST=ITMLST);
: 1622      1737 2     IF NOT .STATUS
: 1623      1738 2     OR (.SUBPROCPID EQL 0)
: 1624      1739 2     THEN RETURN SIGNAL(MAIL$_NOPARENT);
: 1625      1740 2 END
: 1626      P 1741 2 ELSE BEGIN
: 1627      1742 2     $ITMLST_INIT(ITMLST=ITMLST,
: 1628      1743 2     (ITMCOBJ=JPI$_PID, BUFADR=SUBPROCPID));
: 1629      1744 2
: 1630      P 1745 2     IF _ERR($GETJPIW(PRCNAM=COMMAND_DESC, ITMLST=ITMLST));,
: 1631      1746 2     RETURN SIGNAL(.STATUS));
: 1632      1747 2 END;
: 1633      1748 2
: 1634      1749 2 IF .SUBPROCPID EQL 0
: 1635      1750 2 THEN RETURN SIGNAL((SS$_NONEXPR AND NOT STS$_SEVERITY) OR STS$_WARNING);
: 1636      1751 2
: 1637      1752 2 MAIL$DISABLE_CTRLC();
: 1638      1753 2 PURGE_VEC[0] = 0;
: 1639      1754 2 PURGE_VEC[1] = %X'7FFFFFFF';
: 1640      1755 2 $PURGE(INADR=PURGE_VEC);
: 1641      P 1756 2 IF _ERR(LIB$ATTACH(SUBPROCPID));,
: 1642      P 1757 2     (MAP STATUS : $BLOCK;
: 1643      P 1758 2     STATUS[STS$_SEVERITY] = STS$_ERROR;
: 1644      P 1759 2     MAIL$ENABLE_CTRLC();
: 1645      1760 2     RETURN SIGNAL(.STATUS)););
: 1646      1761 2
: 1647      1762 2 MAIL$ENABLE_CTRLC();
: 1648      1763 2 MAIL$CMD_SHOW_NEW();
: 1649      1764 2
: 1650      1765 2 RETURN TRUE
: 1651      1766 1 END;
```

				.EXTRN	SYSS\$GETJPIW	
			003C 00000	.ENTRY	MAIL\$CMD ATTACH, Save R2,R3,R4,R5	1711
55	000000000'	00	9E 00002	MOVAB	COMMAND_DESC, R5	
54	000000000G	00	9E 00009	MOVAB	MAIL\$ENABLE_CTRLC, R4	
53	000000000G	00	9E 00010	MOVAB	SYSS\$GETJPIW, R3	
5E		1C	C2 00017	SUBL2	#28, SP	
		55	DD 0001A	PUSHL	R5	1727
	ECAA	CF	9F 0001C	PUSHAB	SD_P1	
000000000G	00	02	FB 00020	CALLS	#2, CLIS\$GET_VALUE	
	ECAF	CF	9F 00027	PUSHAB	SD_PARENT	1729
000000000G	00	01	FB 0002B	CALLS	#1, CLIS\$PRESENT	
2E		50	E9 00032	BLBC	PARENT_FLAG, 2\$	1731
50		6E	9E 00035	MOVAB	ITMLST, \$\$ITMBLKPTR	1734
80	03030004	8F	D0 00038	MOVL	#50528260, (\$\$ITMBLKPTR)+	
80	18	AE	9E 0003F	MOVAB	SUBPROCPID, (\$\$ITMBLKPTR)+	
		80	7C 00043	CLRQ	(\$\$ITMBLKPTR)+	
		7E	7C 00045	CLRQ	-(SP)	1736
		7E	D4 00047	CLRL	-(SP)	
	0C	AE	9F 00049	PUSHAB	ITMLST	
		7E	7C 0004C	CLRQ	-(SP)	
		7E	D4 0004E	CLRL	-(SP)	
63		07	FB 00050	CALLS	#7, SYSS\$GETJPIW	
05		50	E9 00053	BLBC	STATUS, 1\$	1737
	18	AE	D5 00056	TSTL	SUBPROCPID	1738
		2D	12 00059	BNEQ	3\$	
	007EDF90	8F	DD 0005B	PUSHL	#8314768	1739
		67	11 00061	BRB	5\$	
50		6E	9E 00063	MOVAB	ITMLST, \$\$ITMBLKPTR	1743
80	03190004	8F	D0 00066	MOVL	#51970052, (\$\$ITMBLKPTR)+	
80	18	AE	9E 0006D	MOVAB	SUBPROCPID, (\$\$ITMBLKPTR)+	
		80	7C 00071	CLRQ	(\$\$ITMBLKPTR)+	
		7E	7C 00073	CLRQ	-(SP)	1746
		7E	D4 00075	CLRL	-(SP)	
	0C	AE	9F 00077	PUSHAB	ITMLST	
		55	DD 0007A	PUSHL	R5	
		7E	7C 0007C	CLRQ	-(SP)	
63		07	FB 0007E	CALLS	#7, SYSS\$GETJPIW	
04		50	E8 00081	BLBS	STATUS, 3\$	
		50	DD 00084	PUSHL	STATUS	
		42	11 00086	BRB	5\$	
	18	AE	D5 00088	TSTL	SUBPROCPID	1749
		07	12 0008B	BNEQ	4\$	
7E	08E8	8F	3C 0008D	MOVZWL	#2280, -(SP)	1750
		36	11 00092	BRB	5\$	
000000000G	00	00	FB 00094	CALLS	#0, MAIL\$DISABLE_CTRLC	1752
	10	AE	D4 0009B	CLRL	PURGE_VEC	1753
14	AE 7FFFFFFF	8F	D0 0009E	MOVL	#2147483647, PURGE_VEC+4	1754
	10	AE	9F 000A6	PUSHAB	PURGE_VEC	1755
000000000G	00	01	FB 000A9	CALLS	#1, SYSS\$PURGWS	
	18	AE	9F 000B0	PUSHAB	SUBPROCPID	1760
000000000G	00	01	FB 000B3	CALLS	#1, LIB\$ATTACH	
	52	50	D0 000BA	MOVL	R0, STATUS	
	12	52	E8 000BD	BLBS	STATUS, 6\$	
	00	02	F0 000C0	INSV	#2, #0, #3, STATUS	

MAIL\$CMDS
V04-000

VAX/VMS PERSONAL MAIL UTILITY COMMANDS

N 10
16-Sep-1984 00:47:25
14-Sep-1984 12:42:16

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[MAIL.SRC]CMDS.B32;1

Page 77
(31)

	64	00	FB	000C5	CALLS	#0, MAIL\$ENABLE_CTRL	:
		52	DD	000C8	PUSHL	STATUS	:
00000000G	00	01	FB	000CA	CALLS	#1, LIB\$SIGNAL	:
			04	000D1	RET		:
	64	00	FB	000D2	CALLS	#0, MAIL\$ENABLE_CTRL	:
00000000G	00	00	FB	000D5	CALLS	#0, MAIL\$CMD_SHOW_NEW	:
	50	01	DO	000DC	MOVL	#1, R0	:
			04	000DF	RET		:

1762
1763
1765
1766

; Routine Size: 224 bytes, Routine Base: \$CODE\$ + 143E

MAIL\$CMDS
V04-000

VAX/VMS PERSONAL MAIL UTILITY COMMANDS

B 11
16-Sep-1984 00:47:25
14-Sep-1984 12:42:16

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[MAIL.SRC]CMDS.B32;1 Page 78
(32)

: 1653
: 1654
1767 1 END
1768 0 ELUDOM

.EXTRN LIB\$SIGNAL

PSECT SUMMARY

Name	Bytes	Attributes
\$GLOBALS	4	NOVEC, WRT, RD, NOEXE, NOSHR, LCL, REL, CON, NOPIC, ALIGN(2)
\$OWNS	232	NOVEC, WRT, RD, NOEXE, NOSHR, LCL, REL, CON, NOPIC, ALIGN(2)
\$CODES	5406	NOVEC, NOWRT, RD, EXE, NOSHR, LCL, REL, CON, NOPIC, ALIGN(2)

Library Statistics

File	----- Total	Symbols Loaded	----- Percent	Pages Mapped	Processing Time
_\$255\$DUA28:[SYSLIB]STARLET.L32;1	9776	50	0	581	00:00.8
_\$255\$DUA28:[MAIL.OBJ]MAILDEF.L32;1	457	26	5	26	00:00.2

COMMAND QUALIFIERS

: BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS\$:CMDS/OBJ=OBJ\$:CMDS MSRC\$:CMDS/UPDATE=(ENH\$:CMDS)

: Size: 4704 code + 938 data bytes
: Run Time: 00:47.7
: Elapsed Time: 02:54.7
: Lines/CPU Min: 2222
: Lexemes/CPU-Min: 23343
: Memory Used: 250 pages
: Compilation Complete

0228 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

